Design • Build • Program • Learn • Compete

SERVO MAGAZINE

FOR THE ROBOT INNOVATOR

www.servomagazine.com

November 2015

ALAN

Adaptive **L**earning **A**ndroid **N**ode

25 years of film industry robotics meld art and technology into a customizable kit.

BUILD A BETTER BOT BRAIN

USING PNEUMATICS

PERIMETER SENSING OPTIONS

WHY ARE ROBOTS SO HARD TO DEVELOP?


PCBCART
High-Value PCB Fabrication &
Assembly Services from PCBCART
Save 15%
up to $200 off on your first order
USE CODE
PCBSERVO
We offer:
PCB fabrication up to 32 layers
Min. tracing/spacing to 3mil/3mil
Min. microvias to 0.1mm
Special PCBs-Aluminum, flex and HDI
Full turnkey PCB assembly
Prototype to large production
ISO 9001
LEAD FREE
www.pcbcart.com
sales@pcbcart.com

SERVO MAGAZINE

11.2015
VOL. 13 NO. 11

Columns

PAGE 07

07 Robytes
by Jeff and Jenn Eckert
Stimulating Robot Tidbits
- Sucking Up Flies
- LOCUST Swarm Coming
- Origami Bot Folds Itself
- Stay at the Weird
- USA vs. the Rising Sun?

10 GeerHead
by David Geer
ALAN, the Android Kit
This modular kit to make your own personalized android head uses off-the-shelf components, and can be shared with a wide field of builders, academics, and robot enthusiasts.

74 Then and Now
by Tom Carroll
Why are Robots so Hard to Develop?
Between limitations in technology and a misperceived concept by the public at large about what robots should be able to do is the core frustration of almost every robot builder.

PAGE 74

Departments

06 Mind/Iron
Robotic Engineers vs. Experimentalists

14 New Products

20 Events Calendar

20 Showcase

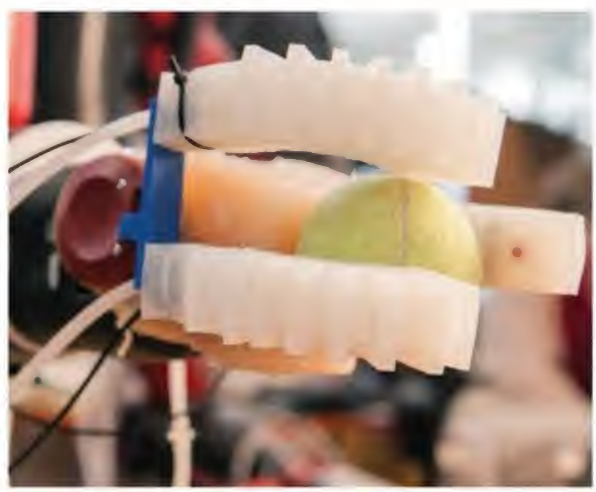
64 SERVO Webstore

80 RoboLinks

80 Advertiser's Index

16 Bots in Brief
- Out of the Frying Pan, Into the Fire
- Harvard RoboBee All the Buzz
- Bend It Like Silicone
- Jibo's Got the Moves
- Marking Your Turf

37 MaxRoboTech Comics
Game Show AI and Robotics



SERVO Magazine **(ISSN 1546-0592/CDN Pub Agree#40702530)** is published monthly for $24.95 per year by T & L Publications, Inc., 430 Princeland Court, Corona, CA 92879. PERIODICALS POSTAGE PAID AT CORONA, CA AND AT ADDITIONAL ENTRY MAILING OFFICES. POSTMASTER: Send address changes to **SERVO Magazine, P.O. Box 15277, North Hollywood, CA 91615** or Station A, P.O. Box 54, Windsor ON N9A 6J5; cpcreturns@servomagazine.com

In This Issue ...

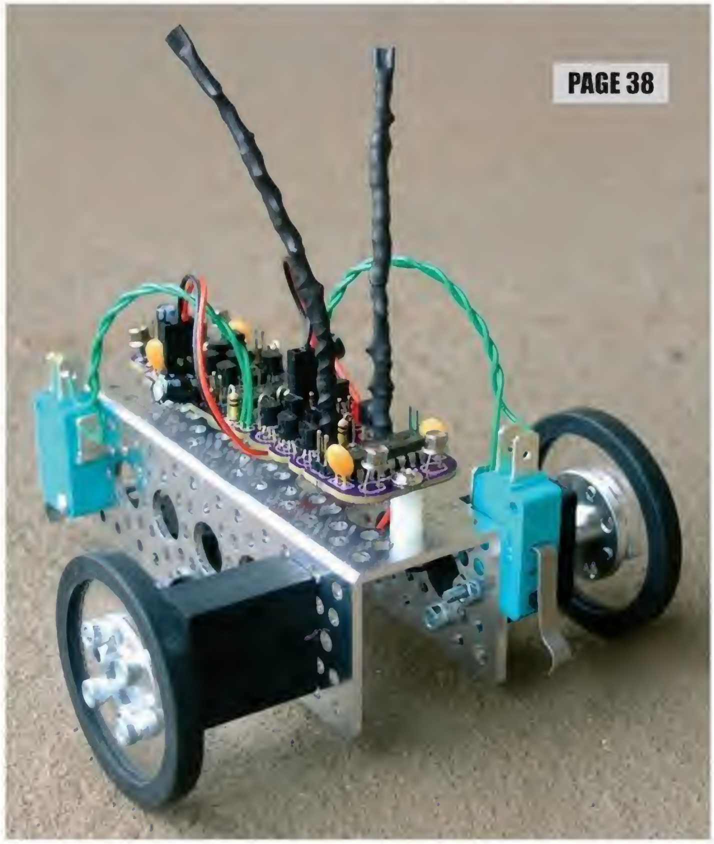
38 Microcerebrum — Build a Better Bot Brain
by Dave Prochnow
Here's a proposal for an inexpensive complete robot controller alternative that can deliver equivalent functionality of programmed control without the coding chore overhead.

47 Animatronics for the Do-It-Yourselfer
by Steve Koci
A Breath of Fresh Air
The use of pneumatics provides prop builders with a unique set of qualities that are often harder to achieve with other methods of movements.

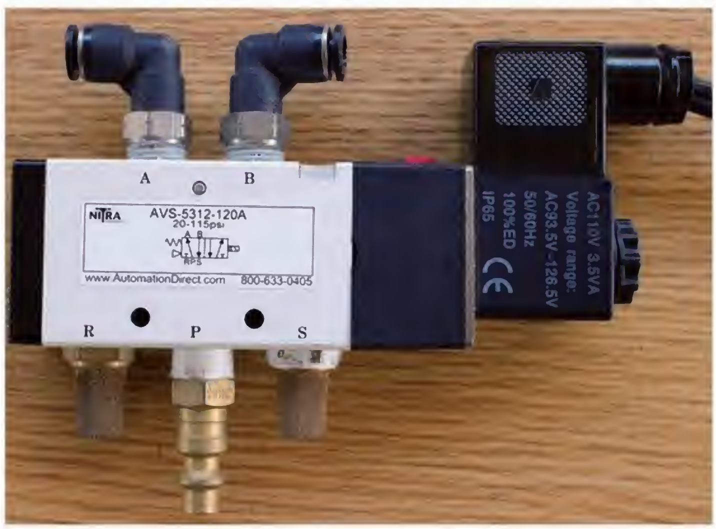
56 *REVIEW:* Mantis The Latest Offering from ServoCity
by Jeff Cicolani
Take to the "off" road with this latest member of the ServoCity robotics platform family.

60 Perimeter Sensing Options
by John Blankenship
Understanding the limitations and advantages of various choices of perimeter sensing can help improve the functionality of your robot.

68 BASIC Bots & PICAXE Processors — Part 3
by Eric Ostendorff
Analog Sensors, ADCs, and a Walking Robot!
Take a look at some simple sensors to give our breadboard the ability to measure distance, light, and temperature, then turn that breadboard into a simple robot.



PAGE 38



PAGE 47

22 BUILD REPORT
BaneBots P60 Repair and a Brushless Upgrade

26 PARTS IS PARTS
Product Beta Test: RageBridge 2 from Equals Zero Designs

28 Big Power, Little Package: Brushless Motor Drive

30 PROFILE
He Caught It, and I Don't Want to Cure It: Robot Fever

32 BattleBots Gets Tombstoned

35 Tips from the Pros

The Combat Zone

Mind / Iron

by Bryan Bergeron, Editor

Robotic Engineers vs. Experimentalists

The innovators who are responsible for propelling robotics to the next level fall into one of two camps: robotic engineers or robotic experimentalists. Robotic engineers leverage the scientific methods learned in academia to design, build, and use robots and robotic systems. That's a tall order for a single innovator, given that robotic engineering requires fluency in all of the engineering domains — from software and electrical, to mechanical and often biomedical engineering. On the other hand, robotic experimentalists leverage hands-on practical experience in designing and developing next-generation robotics.

In my experience, a shop with both experimentalists and engineers is optimally positioned not only to make the major breakthroughs, but to simply get things done. For example, I'm fortunate to work with a robotics lab staffed by both engineers and experimentalists. Recently, in one robot design requiring an onboard reservoir of water-based liquid, the question was posed whether internal waterproof casing and cables were required, should some of the liquid spill internally. The downside of such a provision — added weight, size, and cost — was significant, but so was the cost of replacing a robot.

The question to answer was whether a leak would cause problems with exposed microcontroller boards. If so, then waterproof plastic enclosures would have to be developed for the boards. The engineers in the group discussed — for hours — methods of determining the resistivity of the fluid, from examining the ingredients to using various testing devices. One of the experimentalists in the group simply took a microcontroller board, dunked it in a container of the fluid for a few seconds, wiped it down, and then applied power. Question answered. There was no immediate issue, but over several days some of the connector pins started to oxidize. So, we went with modest waterproofing of the boards and connectors, and inserted a fluid alarm in the body cavity of the robot.

Sure, the direct experimental approach cost a board, but it also answered questions not raised by the engineering group such as what were the effects of rosin and other deposits on the circuit board on conductivity — in the way that water is an insulator until it picks up salt from the environment.

In another instance, the group was debating the best battery chemistry to use with the robot. Because we didn't have the batteries on hand, the discussion was based on published specifications for various battery packs. The engineers quickly calculated maximum discharge and charging rates for given ambient temperatures, and decided on the most appropriate battery chemistry: a new variant of Lithium-Ion. We have yet to test the batteries on real world circuits — a task for the experimentalists.

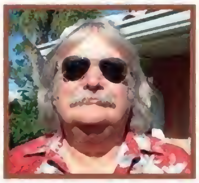
The intuitive grasp of robotics developed by the experimentalists turns out to be optimum when there is product in hand to work with. The engineers are best at applying mathematics to product specifications and then ordering only what they know will work.

Clearly, both the hands-on intuitive perspective and skill set of the experimentalist and the reasoned methodical analysis of the engineer are needed to solve different types of design and development challenges. Although rare, it's possible for one person to develop skill sets in both camps through formal education and lots of hands-on work. If you're passionate about robotics, I challenge you to become one of these hybrid super developer/inventors. SV

FOR THE ROBOT INNOVATOR SERVO MAGAZINE

Published Monthly By
T & L Publications, Inc.
430 Princeland Ct., Corona, CA 92879-1300
(951) 371-8497
FAX **(951) 371-3052**
Webstore Only **1-800-783-4624**
www.servomagazine.com

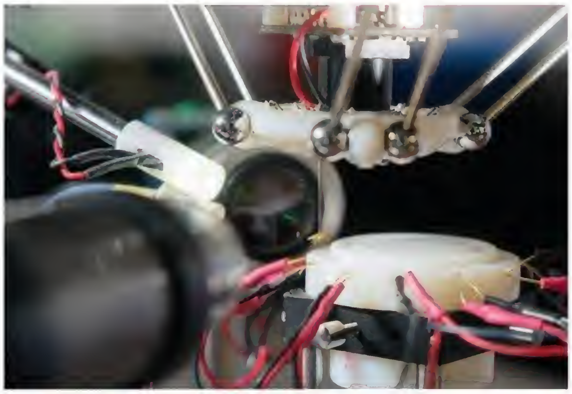
Subscriptions
Toll Free **1-877-525-2539**
Outside US **1-818-487-4545**
P.O. Box 15277, N. Hollywood, CA 91615




Robytes

by Jeff and Jenn Eckert

Sucking Up Flies



Fruit fly hangs at end of robot's suction tube. Courtesy of Stanford.

If you attract a lot of flies, it could be just a basic hygiene problem. If you do it intentionally, however, it's more likely that you are doing genetics research of some sort. In fact, for a century or so, fruit fly research has been at the heart of many scientific discoveries since the critter's short life span allows us to study many generations in a short time, and also because they share 75 percent of the genes that are involved with disease in humans. Until recently, it was necessary to conk them out by cooling or drugging them, and then spending many tedious hours examining dishes of anesthetized bugs. Fortunately, Prof. Mark Schnitzer with Stanford's Bio-X institute (**biox.stanford.edu**) has come up with a robot that can inspect and experiment with totally awake flies.

"Robotic technology offers a new prospect for automated experiments and enables fly researchers to do several things they couldn't do previously," Schnitzer said. "For example, it can do studies with large numbers of flies inspected in very precise ways." The group did one study of 1,000 flies in 10 hours, which would have taken much longer for any human.

When it's time to pick up a subject, the robot hits a container of flies with a flash of IR light to establish their locations, simultaneously analyzing each fly's thorax so it can be individually recognized. It then sucks up a specimen using a tiny suction tube (painlessly, we're told, although no individual flies have confirmed that). It can even carry out a microdissection and take a look at the fly's brain.

The bottom line is that it will now require less human drudgery when unraveling the mysteries of human aging, cancer, diabetes, and a range of other diseases.

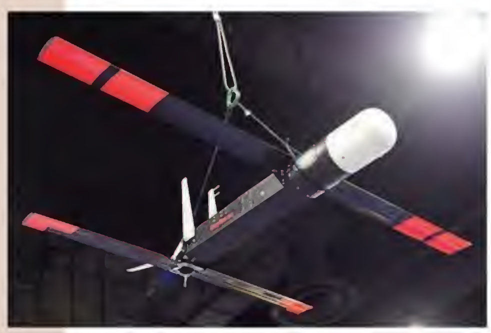
LOCUST Swarm Coming

LOCUST (l) leaving launcher and (r) wings unfolding.

Another insect-inspired concept is the Low Cost UAV Swarming Technology (LOCUST), recently demonstrated by the Office of Naval Research (**www.onr.navy.mil**). According to ONR officials, LOCUST can launch swarming UAVs (unmanned aerial vehicles) to "autonomously overwhelm an adversary. The deployment of UAV swarms will provide Sailors and Marines a decisive tactical advantage."

The program includes a tube-based launcher that can rapidly send 30 UAVs into the air. The technology then utilizes information sharing between the UAVs, enabling autonomous collaborative behavior in both defensive and offensive missions. The Navy was typically non-specific in terms of weaponry, but it did note that the UAVs can carry "varying payloads for different missions." Also noted was that although this is far more cutting-edge technology than remote controlled vehicles, there will always be a human monitoring the mission and able to take control whenever desired.

The low cost aspect of the program is particularly important, and hundreds of these vehicles can be deployed at a price lower than even one tactical aircraft. This will force adversaries to focus on a highly complex swarm response.



LOCUST displayed at 2015 Sea-Air-Space Expo.

Origami Bot Folds Itself

MIT's origami bot walks, climbs, and swims using external actuation.

Even if your childhood interest in origami never progressed much beyond paper airplanes and flappy birds, you may be impressed by an origami robot developed at MIT's Department of Electrical Engineering and Computer Science (**www.eecs.mit.edu**). The contraption — demonstrated at the IEEE International Conference on Robotics and Automation — starts out as a printable, flat sheet of plastic that folds itself up when exposed to heat. Powered via a permanent magnet attached to its back, the assembled robot (weighing only 1/3 of a gram and measuring about a centimeter in length) can swim, climb inclines, navigate through rough terrain, and carry twice its weight.

Unlike other origami robots that require electronics and motors for actuation, this one is controlled entirely by external magnetic fields which cause the body to flex and twist so as to move it forward at nearly four body lengths per second.

According to MIT, the robot's design was motivated by a hypothetical application in which "tiny sheets of material would be injected into the human body, navigate to an intervention site, fold themselves up, and, when they had finished their assigned tasks, dissolve." The researchers therefore built prototypes from liquid-soluble materials.

One prototype dissolved almost entirely in acetone (everything but the magnet), whereas another had water-soluble components. "We complete the cycle from birth through life, activity, and the end of life," noted researcher Shuhei Miyashita. "The circle is closed."

The Henn-na Hotel features robotic receptionists, porters, room service, and others.

Stay at the Weird

If you're planning a trip to the southern Nagasaki Prefecture (Japan) and are up for a weird experience, check out the Henn-na Hotel (**www.h-n-h.jp**). And we don't use the term "weird" disparagingly; "henn-na" actually means "weird." For one thing, the Henn-na is the world's first hotel attempting to cut costs by using a staff made up almost entirely of robots (they still haven't found one that can make beds), so there are no more than 10 live bodies working there at any given time. For some unexplained reason, the English-speaking receptionist is a ferocious-looking dinosaur, whereas the Japanese version is a female humanoid that blinks her lashes at you.

The concept becomes even murkier when you discover that the hotel is part of the Huis Ten Bosch — a 380 acre waterfront theme park celebrating Dutch culture (huh?). In any event, standard guest rooms offer 226 ft^2 of floor space, which is quite a bit smaller than the US average of 325 ft^2. However, "superior" (306 ft^2) and "deluxe" (355 ft^2) rooms are also available.

Each room has a hair dryer, electric kettle, and cups and mugs, and Wi-Fi is offered throughout. However, there are no refrigerators or TVs (you can watch on a tablet), and rooms use a "radiant panel" air conditioning system to save money. Yeah, it's different. We might recall the Hunter S. Thompson line, "It never got weird enough for me."

Various travel sites have reported the cost of one night's stay at an unusually cheap 9,000 yen (about $80), but the hotel website specifies 44,280 yen (about $320) for two guests. So, it would be wise to check before making reservations.

Go to **www.servomagazine.com/index.php/magazine/article/november2015_Robytes** to comment on these topics.

The MegaBot Mark 2, gearing up for battle.

USA vs. the Rising Sun?

We are all familiar with robot competitions in the realms of sports and combat, but let's face it. Most are pretty lame, as demonstrated at the pathetic 2015 Robocup soccer tournament. However, in 2012, Japan's Suidobashi Heavy Industries created the Kuratas giant robot, billed as the world's first giant boarding robot. The 4,500 kg vehicle is crewed by a single operator who sits inside the bot's body. It can be armed with such weapons as a 6,000 round-per-minute BB Gatling cannon or a LOHAS launcher that fires things like water bottles. It also has a humanoid hand for picking up things.

Kuratas ruled unchallenged until recently, when the Boston-based MegaBots threw down the gauntlet after introducing MegaBot Mark 2: a six ton unit piloted by a team of two. Reportedly, the Mark 2 can fire three pound paint cannonballs at up to 100 MPH, and its Gatling guns feature an advanced targeting system.

"Suidobashi, we have a giant robot, you have a giant robot; you know what needs to happen," said MegaBots co-founder, Matt Oehrlein in a video while wearing an American flag as a cape. "We challenge you to a duel."

The battle would occur in one year (after both teams have had a chance to make appropriate battle modifications), at a location chosen by Suidobashi. Will the challenge be accepted? We're waiting, sucka! SV

by David Geer
geercom@neo.rr.com

ALAN, the Android Kit

Head and shoulders above the crowd!

William Huff is a roboticist and makeup/special effects (FX) guru who is famous for his movie animatronics puppets and his work in films such as, "The Curious Case of Benjamin Buttons," "Thor," "The Watchmen," "Austin Powers (2 and 3)," and "Master and Commander." It is from this extensive background that "ALAN" comes from. ALAN, the Android kit is the product of 3D modeling and 3D printing technology which will soon be available to enthusiasts and professionals everywhere.

ALAN skinless with the moldings that will help to enable his facial expressions.

"Several kits will be available from a basic low cost shell type kit with no hardware, to fully assembled robots with no paint, and finally fully painted robots that are ready for immediate use by the higher end consumer. This presents users with choices that match their budgets," Huff explained.

ALAN has many layers of technology and creativity so that roboticists can transform the kit using any face or personality they find applicable. Let's look a little closer at what waits inside ALAN.

Location, Station, Preparation

In the modest offices of Robomodix in Portland, OR and their manufacturing arm in Burbank, CA, Huff and his staff of seven (three in the main office and four on hand at the manufacturing plant) design, mold, and handcraft the parts. These are not overly-simplified injection molded parts.

There is a reason Huff avoids this approach. "The process of pressure injection limits many design features that we want to keep in our robots," said Huff.

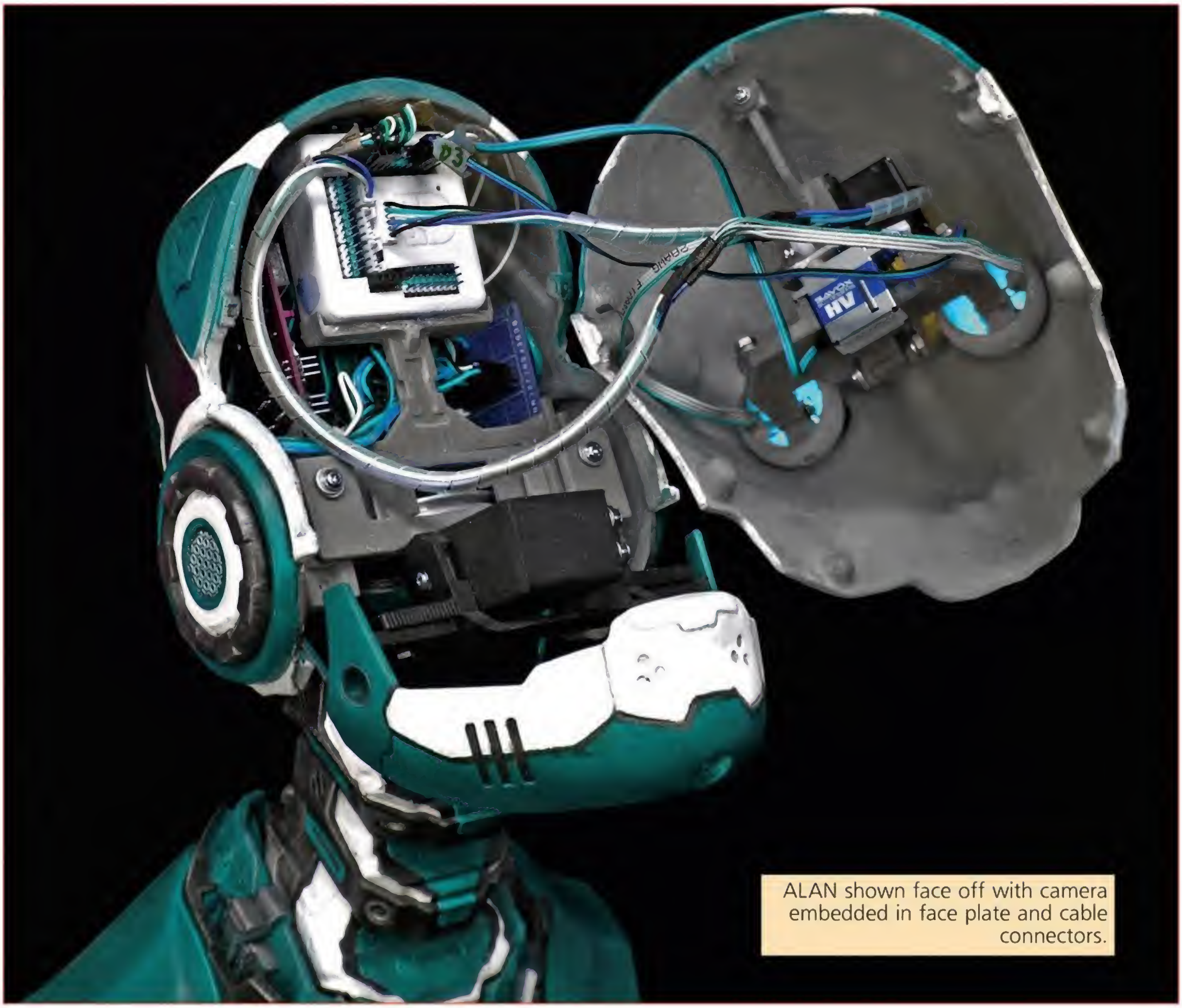
Huff and his team separate themselves from the rest of the robot design crowd in matters of aesthetics, materials, and mechanisms. "It's our opinion that robot aesthetics have not evolved much. Aren't you tired of black and white, smooth surface robots dominating the market?" Huff asked. So, Huff and his crew seek to quickly push new color varieties and intricate design features into the accelerating development of the robot platform.

"Our exposure to robots has been largely through movies and TV, which has developed an underlying subconscious expectation as to how robots should appear. Those are the kinds of robots we want to build," announced Huff.

From 25 years of film industry robotics and make-up experience, Huff and his colleagues are melding art and technology into one. Using components such as silicone skins that are fabricated using 3D designs and 3D printing, they create the same kind of highly detailed look and feel for ALAN.

"Changing or resizing internal parts in software meant that we

Post comments on this article at **www.servomagazine.com/index.php/magazine/article/november2015_GeerHead**.



ALAN shown face off with camera embedded in face plate and cable connectors.

could quickly reprint them, which allowed for fast experimentation," disclosed Huff.

Not All in a Name, but Some

So, where did the name ALAN come from; how did it emerge? After slaving gruelingly over the name and waiting for something magical to appear in his mind, Huff saw the movie, "The Imitation Game," which presented him with Alan Turing. "Turing's thoughts and ideas about AI and machines paralleled what we are trying to develop here. Then came the acronym: A for Adaptive, L for Learning, A for Android, N for Node," Huff shared.

How will ALAN evolve and gain mainstream adoption? How will any android for that matter? Huff believes the key to accepting humanoid robots in everyday social environments will be how they look visually and function. "We had to leap over that uncanny valley (the foreseeable instance where humans reject robots because they are indistinguishable from humans) to make ALAN acceptable as a humanoid. An android is human in its form, and we wanted to stay clear of it looking (too) human," said Huff. Thus explains the obvious absence of hair, skin tones, eye lashes, and the like.

The ALAN android head is modular in its design, and its creator has a lot of planning in the works for performance upgrades including facial expressions. "We even have designs for an entire body for ALAN with some

ALAN components without color, disassembled.

very unique features we have not seen on any other robot!" Huff asserted.

Ultimately, Huff would like to see versions of a fully formed, full bodied ALAN programmed and equipped as a helper robot for the elderly; a household companion; a robot performing social tasks and duties such as reading to and conversing with you; doing Internet searches; providing directions and guidance; and doing other simple daily jobs.

A Science to ALAN

OTS (off-the-shelf) components make it possible to share ALAN with a wide field of builders, academics, robot enthusiasts, and others. "We started our development around EZB by EZ-Robots for the microprocessor and the camera," said Huff.

"Through cloud sharing, EZ-Robots enables users to upload their ALAN projects and share them with other users. This allows users to get up to speed running their ALAN-based projects quickly," commented Huff.

The EZ-Robot camera provides ALAN with visual processing of objects and faces for recognition. "Third-party software like EZ-AI — which is designed to work with EZ-Builder (the EZB software) and Roborealm software — will process the camera data and create databases for uses such as facial recognition recall," explained Huff. This way, ALAN can remember you and know you, associating you with other personal data. "The Roborealm software extends the vision processing capabilities to include shape matching, pupil detection, and object counting," explained Huff.

ALAN also has mechanisms for jaw motion and voice sync activation that a third-party board called the audio servo controller can assist with. "When the incoming audio signal is processed, that signal is translated into servo positions, and we found that we had the most control with the sensitivity through adjusting the onboard potentiometers," Huff revealed.

Chat Robotics

With the EZ-Builder EZB microprocessor — which includes Wi-Fi — you can connect to ALAN and the Internet at the same time. "This is interesting because you have all the capabilities of ALAN, plus the added ability to access external links like ChatBots, web queries, and directions," said Huff.

ChatBots are programs that run on the Internet and appear as if you are conversing with a real person. ALAN can take on these personalities by typing in the ID number of the ChatBot you'd like to use in EZ-Builder.

There are many such ChatBots, including those from Pandorabots.com. "You can also design your own personality with questions and answers from scratch using a program like Program O, which is designed to create ChatBots or to modify existing ones. EZ-AI also comes with a local ChatBot that you can customize," commented Huff.

Resources

Get a Head in Robotics
http://getaheadinrobotics.com

Internet movie database link
to work done by William Huff
www.imdb.com/name/nm0400235

Speech Synthesis, Facial Reconstruction

ALAN achieves speech synthesis, the ability to hear, and respond using Windows environment components that EZ-Builder calls on. This is really computer-generated simulation of speech. "Windows compatible SAPIS voices are available for purchase online, so you can change ALAN's voice," said Huff. Further, the speech recognition enables you to give ALAN voice commands that it can carry out.

You can upgrade ALAN's face with new skins through a magnetic modular system and subskull, which is the area beneath the silicone skin. This enables users to change both genders and faces.

The only requirement for ALAN is a computer or tablet running Windows, if you build your ALAN around the EZB technology mentioned here.

Final Thoughts

ALAN represents just the beginning of Huff's vision for robot enthusiasts interested in androids, and ALAN's development is only limited by your imagination. This is a unique platform to interact with and learn about robotics. You can experience emerging technologies such as ChatBots and AI on a personal level.

Simply put, you can truly get ahead in robotics. **SV**


littleBits
THE ULTIMATE INVENTOR'S TOOLBOX.
For the kids who bust things open to see how they work. The explorers. Inventors. Experimenters. With the littleBits Gizmos & Gadgets Kit, your mischief is your masterpiece. See more at LITTLEBITS.COM/SERVO
GIZMOS & GADGETS KIT
littleBits
GIZMOS & GADGETS
littleBits

NEW PRODUCTS

Actobotics Servo Controller

The Actobotics® Servo Controller is a compact two-channel servo controller that is extremely simple to use, while providing some very unique features. Connecting servos can be done in one of two ways: 1. NEARBY (via Direct Connect). The servos can simply plug into the row of pins on the top of the board just as they would on a receiver or regular servo controller. This option is commonly

used when the servos are located near the controller and there's no need for lengthy wires. 2. LONG DISTANCE (via CAT6). A CAT6 cable can be plugged into the CAT6 port and run out to a CAT6 receiving board; the servos would then plug into the row pins located on the board. This setup is excellent for operating the servos from a long distance, and provides a clean and inexpensive way to extend the servo wires.

While the servo controller is ready to run right out of the box, the PWM (pulse width modulation) signal range can easily be changed in order to tailor the controller to a specific application. The stock PWM range is set to approximately 1,000-2,000 µsec (somewhat standard within the servo industry), which will cause most servos to rotate 90 degrees total (45 degrees each way from center).

To change the range of the servo, the PWM range can be reprogrammed with a single button located on the circuit board. No computer is needed.

The servo controller is capable of handling 4.8V-7.4V nominal voltage which can be provided through the 2.5 x 5.5 mm power jack or into the row pins on the top of the board. The contoured aluminum box provides excellent protection for the board and fits well in the user's hands.

The tapped holes in the back of the box make mounting it to Actobotics channel (or any other component utilizing the standardized .770" hub pattern) simple. This controller comes fully assembled, ready to plug in and put to use. Price is $79.99.

4WD and 6WD Mantis Off-Road Rovers

The Mantis™ 4WD and 6WD rovers also from ServoCity will take off-road robotics excursions to the next level. The bug-like chassis offers extreme A-arm style suspension and nearly 5" of independent wheel travel. With 313 RPM ball-bearing precision planetary gearmotors, there's plenty of torque to turn the aggressive 5.4" off-road tires.

The suspension incorporates 4.62" aluminum beams and 130 mm oil-filled aluminum bodied shocks. An 18" long boxed channel chassis provides a rigid backbone with endless mounting and customization options, making it easy to bolt on Actobotics components and electronics directly to the chassis of the rover.

Rubber grommets and end caps are included for protecting the motor wires as they get routed to the motor controller of choice.

The 2.2" diameter black revolver wheels are driven using 12 mm aluminum hex wheel adaptors to ensure a solid, no slip connection. The 4WD version is priced at $329.99; the 6WD version is $479.99.

For further information, please contact:

ServoCity www.servocity.com

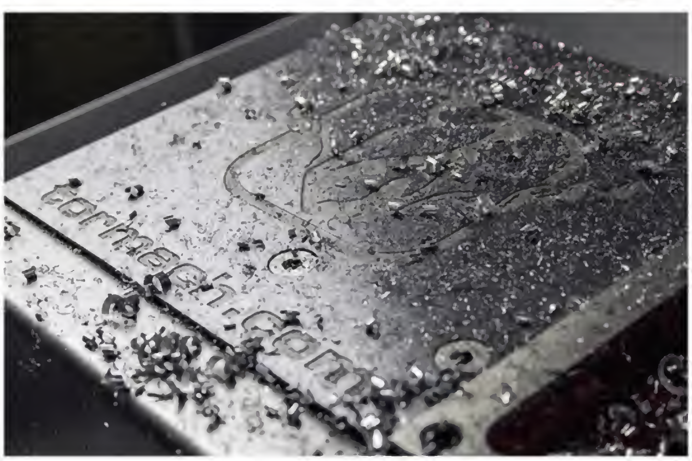
New Milling Machine – the PCNC 440

Tormach, Inc., has announced their newest entry in personal CNC machinery: the PCNC 440. This CNC mill takes the design philosophies of Tormach's other machinery and makes them fit in an even smaller package.

The PCNC 440 has a footprint of 40" x 32" x 42" (X x Y x Z) and weighs just 450 lbs, yet boasts a spindle speed of 300 to 10,000 RPM, a max feed rate of 135 IPM (X,Y), and the ability to cut everything from plastics to aluminum to harder metals like steel and titanium.

It is controlled by PathPilot — Tormach's control system that is used on every PCNC mill.

"This machine truly fits all-around capability in a compact and affordable package. It's perfect for anybody that wants to do real cutting, but doesn't have a lot of space," commented Andrew Grevstad, Tormach's product marketing manager.



Tormach's new PCNC 440 starts at $4,950. PCNC 440s will start shipping in November 2015.

For further information, please contact:

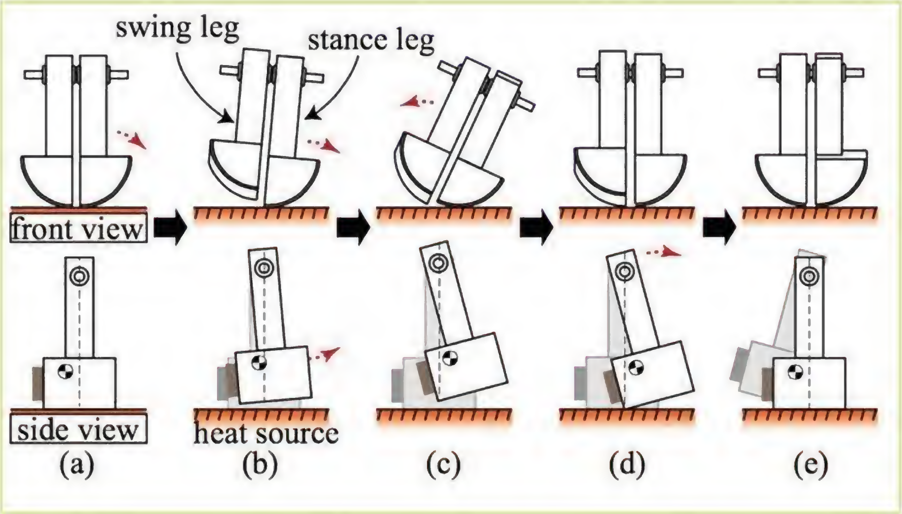
Tormach	www.tormach.com

bots
IN BRIEF

OUT OF THE FRYING PAN, INTO THE FIRE

A bimetal consists of two different kinds of metal stuck together. Bimetals are different than alloys, where the two metals are blended; in a bimetal, the two metals are just layered on top of each other. The reason to do this is to take advantage of the different characteristics of different metals when they're heated. They expand at different rates, causing the piece of bimetal to deform until it cools off again.

Essentially, a bimetal is a way to convert heat directly into mechanical energy, and researchers at the University of Tokyo have come up with a way to leverage this to get a robot to walk. The robot — called Thermobot — has no sensors and no actuators, and as long as it's got a hot surface to walk on, it can keep going pretty much forever.



The trick to this is a combination of two things. First, Thermobot's feet are heavily weighted at the backs, meaning that when the bimetal on one foot deforms and the robot rocks sideways, the foot swings forward. At the same time, the weights on Thermobot's legs control the distance that the robot tips laterally. If everything is calibrated properly, Thermobot will tip back onto the swinging foot when it's ahead of the body and the robot has taken a step.

It's important to note that this technique won't work in an environment that's hot overall. The surface has to be significantly hotter (50 degrees Celsius, ideally) than the rest of the environment in order to give the bimetal a chance to cool so the cycle can repeat itself. In this particular case, Thermobot is walking on a hot plate that's 170 degrees C in a 26 degree room. However, it doesn't matter that much what the absolute temperatures are, just that there's a significant relative difference between them. Thermobot is quite happy to walk in 300 degrees C temperatures as long as the surface is hotter than ambient.

This is certainly one of the more novel walking techniques — especially since it's able to walk on a level surface by harvesting heat. However, no practical applications immediately come to mind. Researchers say they're working on combinations of metals that will allow Thermobot to operate at much lower temperatures — perhaps even safe temperatures — and that it might make for a pretty cool toy. They're also going to be working on "other locomotion mechanisms that can be realized using the self-oscillation of bimetal sheets." Perhaps we'll see more creative robots making use of this tech in the near future.

HARVARD ROBOBEE ALL THE BUZZ

For the last several years, Harvard has been developing a robot bee and they've done some impressive work. Their sub-paper-clip-sized 100 milligram flapping wing micro aerial vehicle is fully controllable down to a stable autonomous hover. It's still tethered for power and there's no onboard autonomous control, but the robot flaps its wings and flies like an insect which is pretty darn cool.

Tiny robotic bugs have lots of potential for search and rescue, surveillance, and exploration, but what's been all the rage recently is adaptive multi-modal robotics: Robots that can creatively handle a combination of terrains, making them much more versatile. With some exceptions, robots are usually pretty bad at this, and with some exceptions, humans and animals are too. There are ground robots that can handle water and a few flying robots that aren't totally helpless on the ground, but so far, we haven't seen much in the way of flying robots that are good swimmers.

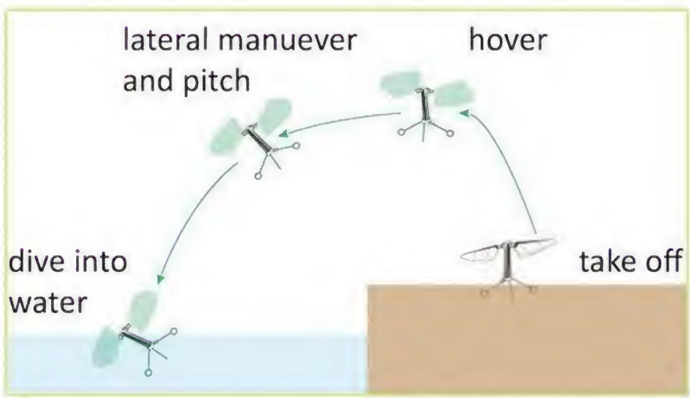
At the recent IROS (Intelligent Robots and Systems) conference, Harvard researchers presented a paper describing how they managed to get their robotic bee to swim, which is not really a thing that even real bees are known for doing. With no hardware modifications at all, Harvard's RoboBee can fly through the air, crash land in the water, and turn into a little submarine.

RoboBee is small enough to sit on the tip of your finger, and light enough that you'd barely feel it if it was. When it flies (or swims), it's doing so under full control. A motion capture system tracks its position and sends trajectory commands to the robot. This works in both air and water, and RoboBee's method of entry (a pitch-over, dive, crash, and sink) is deliberate.

The key realization here is that swimming is actually a lot like flying. In both cases, you're trying to propel yourself through a fluid by moving a wing (or fin) back and forth. To fly (and particularly to hover), you need to do this very quickly, but to swim it's a much more relaxed motion. Fundamentally, it's the same motion, though, and you can achieve it with the same basic hardware. In the case of RoboBee, to fly in air, it flaps its wings at 120 Hz, while to swim in water, it flaps its wings at just 9 Hz. Otherwise, the three-axis torque control is very similar, meaning that the robot can be steered around in the water, too.

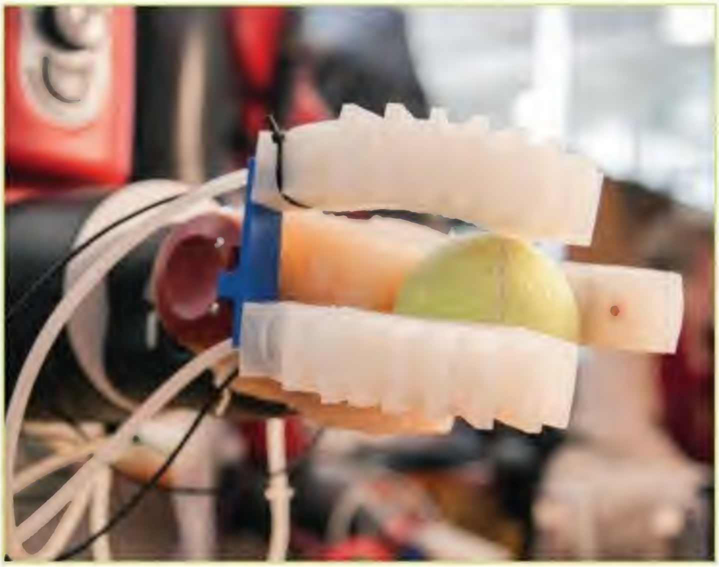
One unique problem that RoboBee has with the water entry is that it's so small, the surface tension of the water is enough to keep it from submerging. This is part of the reason that it has to crash land in water (it also needs to have its wings treated with a surfactant to help it sink).

Images courtesy of Wyss Institute/ Harvard University.



A fully loaded RoboBee (with a battery on board) might be heavy enough to avoid this problem, but at this point it's still an issue. Also still an issue is the whole water-air transition, which seems like it's significantly more difficult than going from air to water, but we've been assured that the researchers will be tackling this in future work.

BEND IT LIKE SILICONE



Robots have many strong suits, but delicacy traditionally hasn't been one of them. Rigid limbs and digits make it difficult for them to grasp, hold, and manipulate a range of everyday objects without dropping or crushing them.

Recently, researchers from MIT's Computer Science and Artificial Intelligence Laboratory (CSAIL) have discovered that the solution may be to turn to a substance more commonly associated with new buildings and Silly Putty: silicone. At a recent conference, researchers from CSAIL Director Daniela Rus' Distributed Robotics Lab demonstrated a 3D printed robotic hand made out of silicone rubber that can lift and handle objects as delicate as an egg and as thin as a compact disc.

Just as impressively, its three fingers have special sensors that can estimate the size and shape of an object accurately enough to identify it from a set of multiple items.

"Robots are often limited in what they can do because of how hard it is to interact with objects of different sizes and materials," Rus says. "Grasping is an important step in being able to do useful tasks; with this work, we set out to develop both the soft hands and the supporting control and planning systems that make dynamic grasping possible."

The gripper — which can pick up such items as a tennis ball, a Rubik's cube, and a Beanie Baby — is part of a larger body of work out of Rus' lab at CSAIL aimed at showing the value of so-called "soft robots" made of unconventional materials such as silicone, paper, and fiber.

Researchers say that soft robots have a number of advantages over "hard" robots, including the ability to handle irregularly-shaped objects, squeeze into tight spaces, and readily recover from collisions.

"A robot with rigid hands will have much more trouble with tasks like picking up an object," graduate student, Bianca Homberg said. "This is because it has to have a good model of the object and spend a lot of time thinking about precisely how it will perform the grasp."

Soft robots represent an intriguing new alternative. However, one downside to their extra flexibility (or "compliance") is that they often have difficulty accurately measuring where an object is, or even if they have successfully picked it up at all.

That's where the CSAIL team's "bend sensors" come in. When the gripper hones in on an object, the fingers send back location data based on their curvature. Using this data, the robot can pick up an unknown object and compare it to the existing clusters of data points that represent past objects. With just three data points from a single grasp, the robot's algorithms can distinguish between objects as similar in size as a cup and a lemonade bottle.

Researchers control the gripper via a series of pistons that push pressurized air through the silicone fingers. The pistons cause little bubbles to expand in the fingers, spurring them to stretch and bend.

The hand can grip using two types of grasps: "enveloping grasps," where the object is entirely contained within the gripper; and "pinch grasps," where the object is held by the tips of the fingers.

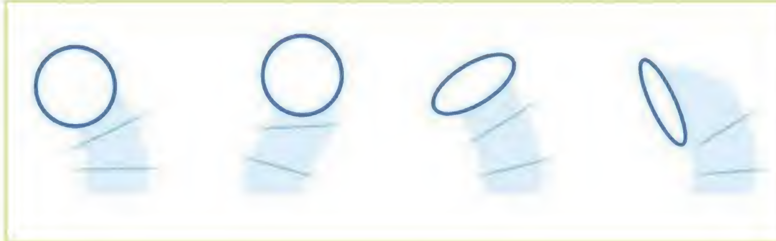
Outfitted for the popular Baxter manufacturing robot, the gripper significantly outperformed Baxter's default gripper, which was unable to pick up a CD or piece of paper, and was prone to completely crushing items like a soda can.

Like Rus' previous robotic arm, the fingers are made of silicone rubber which was chosen because of its qualities of being both relatively stiff, but also flexible enough to expand with the pressure from the pistons. Meanwhile, the gripper's interface and exterior finger molds are 3D printed, which means the system will work on virtually any robotic platform.

In the future, Rus says the team plans to put more time into improving and adding more sensors that will allow the gripper to identify a wider variety of objects.

All of the objects grasped by MIT's soft robotic hand. (Photo courtesy of MIT/CSAIL.)

JIBO'S GOT THE MOVES



Movement is a critical part of Jibo's character. He enjoys 360 degrees of motion thanks to his multi-revolute movement. There are two reasons his design team went this direction. The first is his ability to orient to people all around him, offering full expressiveness regardless of where people are positioned. The second? Honestly? Cynthia Breazeal and her team wanted to be sure Jibo can dance in a really fun and cool way.

When you watch Jibo, he looks like he has a supple spine (line of action) that seems to "bend." He can fluidly interact. There isn't a point where he hits his range of movement and just stops.

The team explored a wide range of concepts before landing on this one. Many were not capable of 360 movement — let alone multi-revolute movement. This design maximizes expressiveness with minimal complexity in terms of degrees of freedom. It is also safe to touch — no pinching joints or points of collision like a swinging arm.

From a social robotics perspective, movement is very important. It ties to how much of human communication is non-verbal. As humans, we express through our bodies all the time. Having a physical robot body capable of expressive movement plays an important role in giving the robot a physical presence in the room.

Jibo was founded by Breazeal, the famed roboticist at MIT's Media Lab and a pioneer of social robotics. Jibo is designed as an interactive companion and helper to families. Breazeal says Jibo is unique because of its emotion and its ability to treat you like a human being. Jibo will come equipped with an initial set of apps that will allow it to play different roles, including a photographer that can track faces and take pictures so you can be in the photos. Jibo can also be an assistant who reminds family members of their schedules.

MARKING YOUR TURF

Have you ever given much thought to how the field marks are painted on soccer and football fields? You know, the marks that designate the sidelines, the penalty box, the circle in the center of the field, etc. The groundsmen who mark these lines need to be accurate. It also takes an incredible amount of time.

Now, there's a robot to do this job.

Intelligent One is being positioned as the first automatic robot for line marking sport fields. The company — which hopes to begin shipping in November 2015 — uses GPS technology to achieve precision of ±2 cm. The robot is powered by a lithium-ion battery and can complete one soccer field in 30 minutes. It can complete 7-8 soccer fields per charge.

Here's the super cool part: The robot can be customized to meet the needs of each particular field and it's all controllable via tablet. The company's pitch is that "every line made is money saved since time, paint, and overall resource consumption are significantly reduced. Intelligent One allows groundsmen to do other tasks while precision is increased and the fields are line marked without supervision."

EVENTS

ROBOTS.NET

Know of any robot competitions I've missed? Is your local school or robot group planning a contest? Send an email to steve@ncc.com and tell me about it. Be sure to include the date and location of your contest. If you have a website with contest info, send along the URL as well, so we can tell everyone else about it. For last-minute updates and changes, you can always find the most recent version of the Robot Competition FAQ at Robots.net: **http://robots.net/rcfaq.html**.

— **R. Steven Rainwater**

NOVEMBER

7 **Bloomington VEX Tournament**
Bloomington, IN
Events include Triad and Full Pull XL.
http://sites.google.com/site/bloomingtonroboticsclub

7 **STHLM Robot Championship**
Stockholm, Sweden
Events include Sumo, Folk Race, Line Following, and Freestyle.
www.robotchampion.se

13-14 **Texas BEST Competition**
Curtis Culwell Center, Garland, TX
Regional for robots built by student teams. This year's event is called Pay Dirt.
www.bestinc.org

20-22 **All Japan MicroMouse Contest**
Atsugi Campus, Tokyo Polytechnic, Atsugi, Kanagawa, Japan
Atsugi Campus, Tokyo Polytechnic, Atsugi, Kanagawa, Japan. Events include MicroMouse Classic, MicroMouse Half-size, and Robotrace.
www.ntf.or.jp/mouse

21 **AHRC Robot Rally**
Pinckneyville Community Center, Norcross, GA
Events include Polyathlon, Mini Sumo, and Line Maze Solving.
www.botlanta.org

22 **International Micro Robot Maze Contest**
Nagoya University, Japan
Events include Micro Robot Racer (1 cm cube), Maze Solver (1 in cube), and Legged Micro Robot.
www2.meijo-u.ac.jp/~ichikawa/MAZEHOME

22 **Robocon**
Tokyo, Japan
University level student robot competition.
www.official-robocon.com

COMBAT ZONE

Post comments on this section and find any associated files and/or downloads at **www.servomagazine.com/index.php/magazine/article/november2015_CombatZone.**

Featured This Month:

22 *BUILD REPORT: BaneBots P60 Repair and a Brushless Upgrade*
by Pete Smith

26 *PARTS IS PARTS: Product Beta Test: RageBridge 2 from Equals Zero Designs*
by Mike Jeffries

28 *Big Power, Little Package: Brushless Motor Drive*
by Russ Barrow

30 *PROFILE: He Caught It, and I Don't Want to Cure It: Robot Fever*
by Matt Leach

32 *BattleBots™ Gets Tombstoned*
by Chris & Tiffany Olin

35 *Tips from the Pros*
by Chris & Tiffany Olin

BUILD REPORT: BaneBots P60 Repair and a Brushless Upgrade

● by Pete Smith

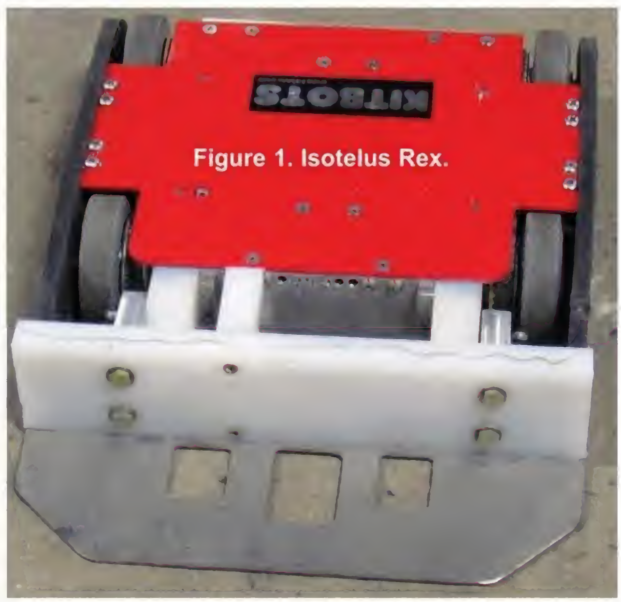


Figure 1. Isotelus Rex.

My 12 lb Hobbyweight, Isotelus Rex (**Figure 1**) lost to Attrition in the finals at this year's Motorama. It wasn't really close, but what forced us to tap out quickly in the final fight was the loss of drive on one side. Discretion being the better part of valor ... it is better to live to fight another day than be carried out of the arena in a dust pan.

I dismantled the bot looking to see what the problem was so that I could rebuild it in time for the Franklin Museum event in October. The only visible damage was to the end plate of the motor on that particular side (**Figure 2**). A big hit must have shocked the armature on the motor and popped out the end plate past the little tabs that are supposed to keep it in place.

When I separated the motor from the gearbox, it was clear why it had seized up. The armature in the motor had moved back far enough so that only a small section of teeth on the

spur gear on the motor were engaging the planet gears in the gearbox. This overloaded the teeth, and they deformed on both the motor (**Figure 3**) and the first stage in the gearbox until it locked up. One of the planet gears can be seen in **Figure 4** next to an undamaged tooth. I dismantled the rest of the gearbox but there was no further impairment.

This damage would have sent most gearboxes into the trash can or the spare pile, but there is a big advantage in using the BaneBots P60 (**www.banebots.com**). You can get replacements for all the various parts. The motor would go in the trash but I already had a replacement for it (**Figure 5**), so all I needed was four new planet gears (less than $8 total plus shipping, saving a lot over the $50+ for a complete new unit).

While I was waiting for those to arrive, I made sure the tabs on the new motor and the other undamaged one were fully formed (**Figure 6**) by bending them over using a flat bladed screwdriver and a few taps with a hammer. This should do the trick, but one could also add a bead of epoxy adhesive around the lip of the end plate to make this failure really unlikely in the future.

When the new gears arrived, I cleaned all the parts and reassembled it with fresh grease (**Figure 7**). This is messy, but not difficult; just note carefully what gear goes where as you take it apart. The motor mounting plate was removed from the old motor and fitted to the new motor. It (and then the motor) and gearbox were attached together with the four long screws to complete the assembly. I now had two good units (**Figure 8**) to go back into Isotelus.

This is where the plans changed.

I had been testing some HobbyKing 45A brushless ESCs (electronic speed controllers) to see how well they worked — both for powering weapons but also for driving the wheels of a 12 or 30 lb bot. I had built a test chassis using a couple of 2848 Tacon brushless motors with

Figure 2. Damage to motor.

Figure 3. Damage to gear.

Figure 5. Spare motor.

Figure 4. Good and bad gears.

Figure 6. Fully formed tab.

Figure 7. Replacement planet gears.


Figure 8. Ready to install.

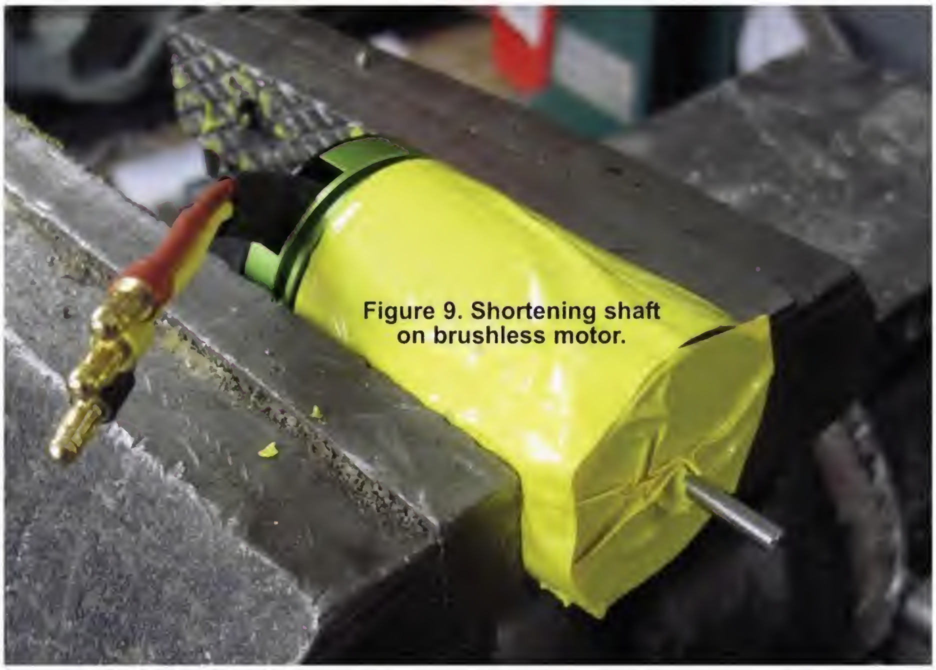


Figure 9. Shortening shaft on brushless motor.

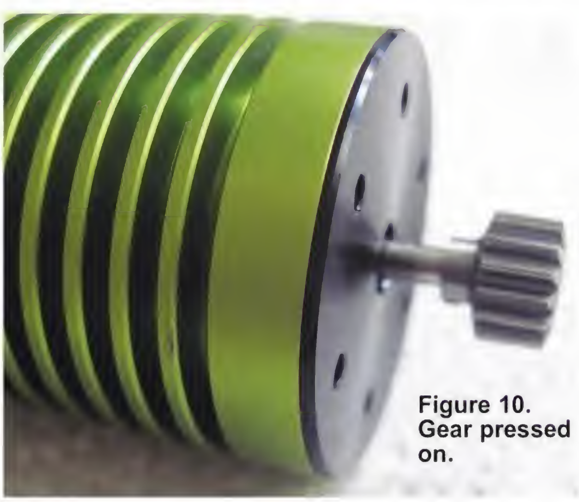


Figure 10. Gear pressed on.

36:1 cordless drill gearboxes. The performance was so good even at only 3S LiPo (11.1V) and on a 26 lb bot, that I thought it might be worth trying brushless motors in Isotelus Rex.

An added incentive was that the Franklin event does not allow LiPo batteries to be used in any bot bigger than 3 lbs (nasty smelling smoke from a LiPo fire is not a good thing, doubly so in a museum that is likely to be hosting a marriage ceremony right after your event!). That leaves using LiFe batteries as the best choice, but the selection of sizes and capacities is much more limited than it is for LiPos. They are also slightly heavier. I had been using a 1,300 mAh 6S LiPo pack and there just didn't seem to be a close replacement in LiFe.

If I could use brushless motors, this would save on the weight, and then I could use a lower cell count but higher capacity LiFe pack and still stay within the weight limits.

I wanted to reuse the BaneBots 16:1 P60s I had, so I would need a motor with a lower kV than those I had used in my test. I settled on the Tacon 540XXL 2858 1,200 kV motor from **www.hobbypartz.com**.

The gear extended out 14 mm from the mounting face on the existing motor, so the first job was to cut down the longer shafts on the Tacons to the same length. I wrapped the motors in tape to protect them (**Figure 9**), and shortened the shafts using a cutoff disk in my Dremel clone. I then cleaned off any burrs left on the end of the shaft with the side of the disk.

The gears are a press-fit, but since the exact dimension of the shaft and the bores of the gears are unknown, I added extra grip by adding a little Loctite 640 in the bores of the gears before pressing them carefully on using my vice (**Figure 10**). This Loctite is designed especially for press-fits and gives a little extra insurance on what already seemed like a pretty good fit.

I had ordered a couple of 500 sized motor mounting plates from BaneBots (the mounting holes are different from the bigger 750 motors I had used before) and the Tacons fit them perfectly (**Figure 11**). They were now refitted to the gearboxes, and the new brushless assemblies were complete (**Figure 12**).

I then rebuilt Isotelus (**Figure 13**), added a HobbyKing CAR-45A brushless ESC for each motor, and did a quick bench test to see that they

worked (the ESCs were programmed with settings 0,0,4,2,3,5,1,1,4 as per the previous tests).

My son, Andrew is designing a new 30 lber and we wanted to see if the same motors and gearboxes could be used in both Isotelus and his new heavier bot. We weighted down the chassis to 26 lb, hooked up a couple of 2S 2,100 mAh Zippy LiFe packs to give 4S, and tried it out. Ughhh! The bot wouldn't move at all!

Must be uncharged batteries, we thought, but even after getting them freshly charged there was still no performance. This was bizarre after the excellent performance during previous tests. We were about to try another battery pack when I noticed I must have had a "senior moment" and had wired the batteries up in parallel not in series, so we had only been running it on 2S!

Correcting that issue, we headed out to try again. This time, the bot accelerated away briskly for about 10 feet and stopped dead! It looked like the speed controllers might be thermally shutting down, but again that would be strange as they had worked well with a bot of about the same weight and performance.

We stripped all the excess weight off and still the bot would only run for a few seconds before stopping. We had taken the top panel off for the last run, and just as it stopped I noticed the LED on the RX started flashing as it stopped. This suggested the problem was with the RX and not the ESCs.

First thought was that the RX was "browning out" due to a voltage drop, but then we noticed the LED would go solid when we approached the bot and start blinking again if we moved away! It seemed to be a radio range issue.

We tried driving the bot in close circles around us and it performed well. Then, I remembered this was the RX that I had used in Trilobite at Clash of the Bots last year and had swapped it out along with the batteries after the bot kept glitching during a fight. The swap had fixed the problem, but I had never got around to tracking down the root cause.

I swapped that RX for a known good one out of one of my hockey bots and all the problems disappeared. The bot performed well — even with all the weights added. The bot accelerates well, is fast, and is very drivable.

One slight problem with all the weights added was a little less braking effect on neutral than would be ideal, but all in all, it was pretty impressive. The faulty RX joined the damaged motors in my trash can.

The new brushless solution is 12 oz lighter than the old Speed 750 drive, and as a bonus the two 2S LiFe batteries are no heavier than the old 6S LiPo packs I used to use.

That means I can use that extra weight to give me more choices in wedge attachments.

Since we can now use these same motors, gearboxes, and ESCs in both our 12 and 30 lb bots, this means we can cannibalize one bot to keep the other going if the drive gets damaged in either.

No more going into a finals match with drive only on one side! SV

Figure 11. Motor fitted to mounting plate.

Figure 12. Completed brushless motor assemblies.

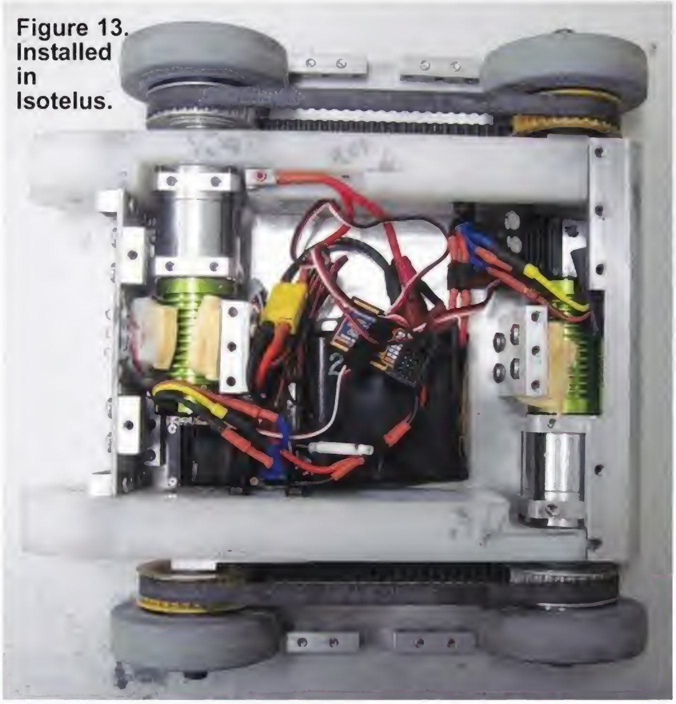


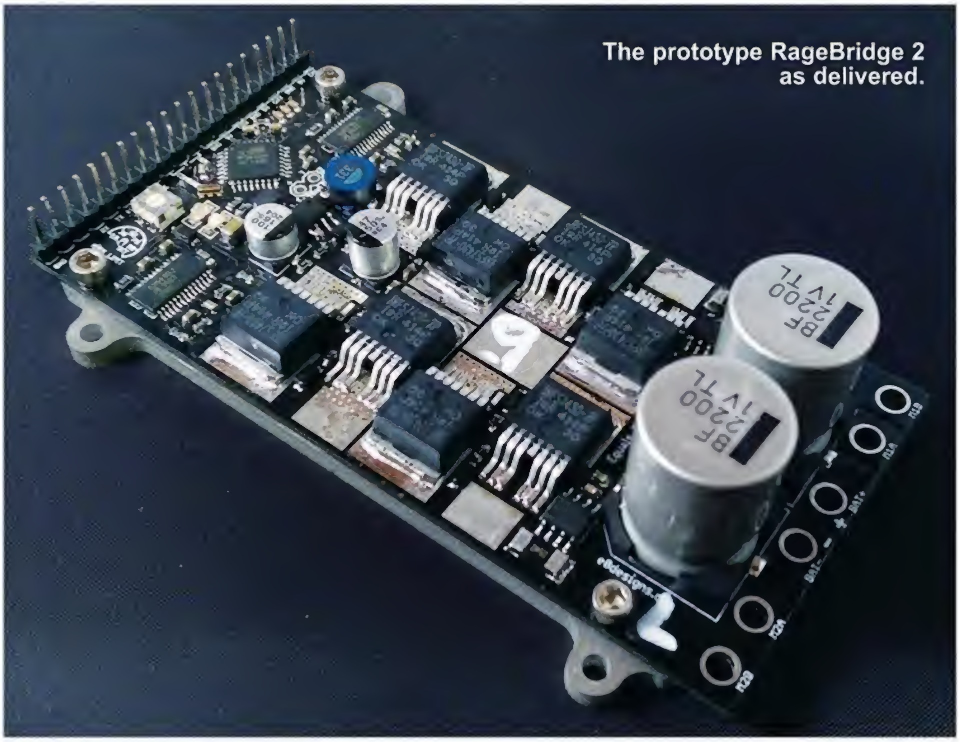
Figure 13. Installed in Isotelus.

PARTS IS PARTS:
Product Beta Test: RageBridge 2 from Equals Zero Designs

● by Mike Jeffries

After a very successful first generation, the RageBridge ESC (electronic speed controller) was redesigned to take advantage of some of the knowledge that stemmed from the first generation, in addition to the integration of some very useful features.

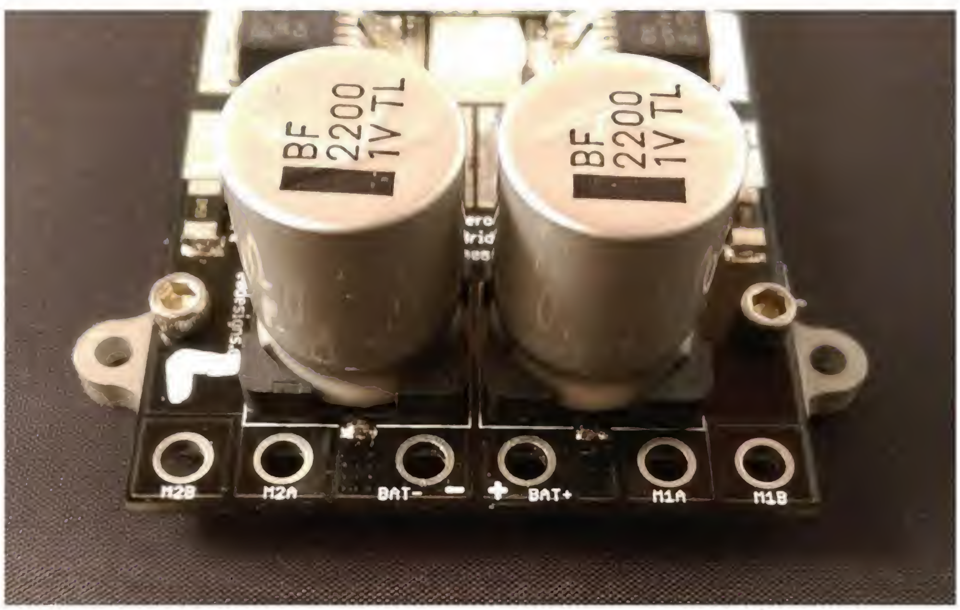
RageBridge 2 Specifications:
Voltage: 8-40 VDC
Amperage: 50A continuous per side
Weight: 4 oz
Dimensions: 4.15" x 2" x 0.9" (without wires)
Price: $199 USD
Special Features: Dual channel motor controller, adjustable current limiting, giga-mode, and constant current mode.



The prototype RageBridge 2 as delivered.

I've been using the first-generation RageBridge since they were in beta testing, and they quickly became my go-to motor controller option for 30 lb combat robots. The throttle response, current limiting, and ease of adjustment combined together to make it stand out from other controllers on the market of a similar scale.

The RageBridge 2 maintains all of the functionality of the original version, while adding some useful extra features and increased power



Power and motor leads are all located in close proximity for easy wiring.

handling capability. The original RageBridge was rated for 30A continuous; the RageBridge 2 is rated for 50A continuous.

On the user side of things, the mixing and current-limiting functions remain as easy to use as ever. In addition to this, the wiring has been simplified by routing the battery power into the speed controller through a single point.

Giga-mode is a standout feature of the RageBridge 2. When activated via a jumper, both sides of the ESC will be combined to provide twice the current handling capability.

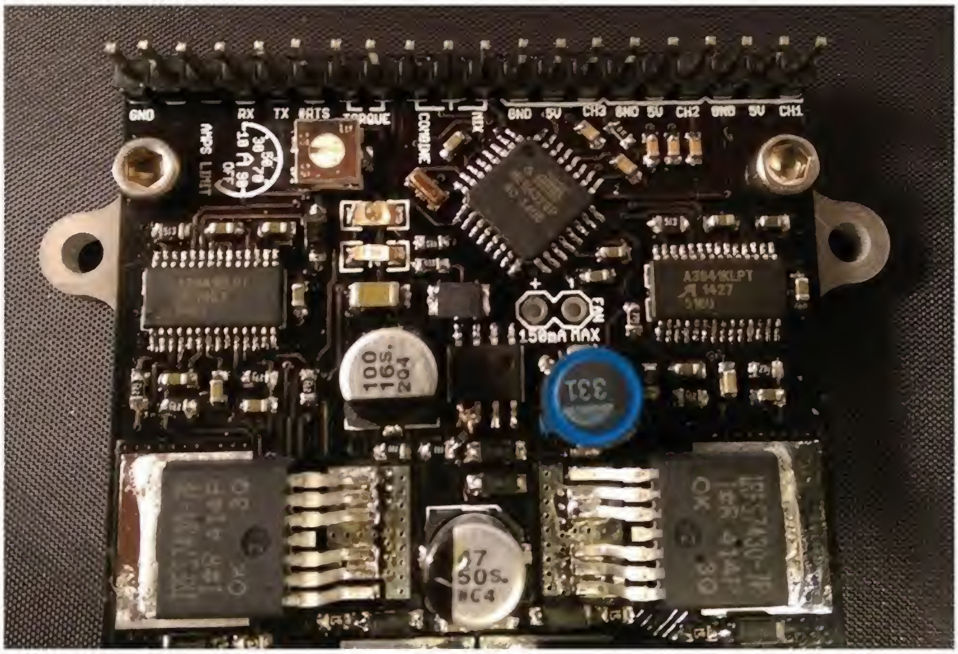
With the original RageBridge, this behavior would work only with specific motors that had multiple isolated brush pairs. With the new RageBridge, the output is synchronized and allows the combined current handling capability to apply to any brushed DC motor.

This extra capacity turns the RageBridge 2 into a monster single channel ESC with a 100A continuous rating and adjustable current limiting up to 180A. This takes the RageBridge 2 from a controller that's a great fit for 12-60 lb combat bots into a controller worth considering for robots weighing up to 250 lbs — so long as the continuous current rating isn't dramatically exceeded.

The test platform I used for the RageBridge 2 was a 12 lb robot called Dolos. Dolos is a low, fast 12 lb bot with power-hungry motors. The RageBridge 2 was tested in competition at Dragon Con Robot Battles and met every expectation I had for it.

The current limiting was put to the test during the competition when one side of the drive system on Dolos locked up in the middle of the match. For the next minute, the motor on that side was completely stalled and getting every bit of current it could take sent to it.

Once the fight was over, the drive system was unjammed and everything continued to work as intended.



Plugs for the PWM cables, jumpers for mode selection, and current adjustment are all located together on one end of the board.

Even if accidental, this was a very important test. If the current-limiting function of the RageBridge 2 wasn't working as intended, the drive motor would have quickly burnt up taking Dolos out of the competition. The current limiting did its job, and after some minor work to un-jam the drive system, Dolos was ready to fight again.

As with the first RageBridge, the RageBridge 2 is a fantastic motor controller with a range of useful features. If the specs of the RageBridge 2 fit your robot's needs, you won't be disappointed by going with it over other controllers in the same size/power range. SV

The prototype RageBridge 2 installed in Dolos.

Big Power, Little Package: Brushless Motor Drive

● by Russ Barrow

Brushless motors are quickly becoming the standard motor for weapon drive in combat robotics. In addition to removing the shock and current-sensitive brushes that wear over time, the brushless motor packs more copper windings and magnets into a given space. Put simply, there's more power per given size. For driving the wheels of a robot, brushless drive seems to be a dream just out of reach ... but no longer as two solutions have come together to provide a reliable and controllable drive platform in the smallest of space.

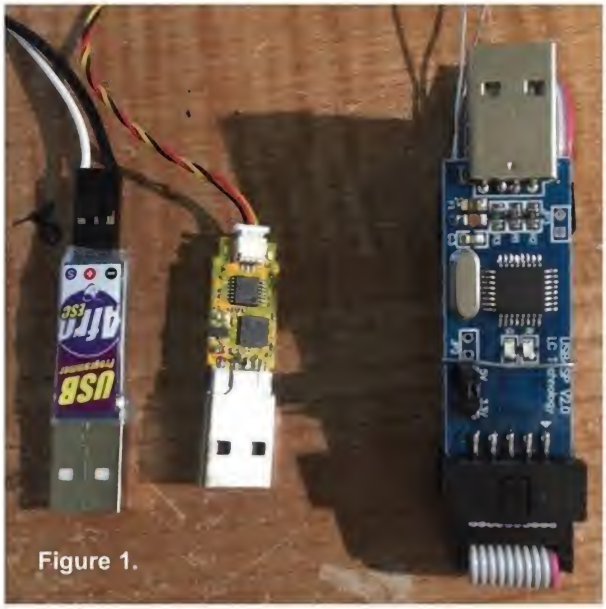


Figure 1.

As brushed motors are mechanically commutated to generate rotation given a voltage, brushless motors require a drive or ESC (electronic speed controller) that can start and run the brushless motor by controlling power through the motor's three copper coils. Many small or inexpensive brushless motors lack feedback Hall-effect sensors that provide information on the direction and speed of the spinning rotor. To control these motors, ESCs must sense the changes in field voltage made by the rotation of the magnetic rotor inside the coils.

Most ESCs use firmware (hardware-specific software) to start the motor at a relatively moderate speed to get the rotor spinning, and adjust direction and speed to "lock in" the rotor synchronously (at the same time) with the change in coil magnetic fields. This type of control yields poor low speed performance, but is the basis of most airplane or helicopter ESCs. For aircraft, low speed is not necessary as the propellers used do not generate much lift at low speeds.

Figure 2.

The demand for aerobatic flight controls has changed the nature of the startup procedure of brushless ESCs. Several software developers have created very sophisticated sensing and low speed timing code for several microcontroller platforms. For Atmel-based microcontrollers, one of the more popular firmware sets is called SimonK, as it was developed by a Canadian engineer by the name of Simon Kirby. Another popular firmware set is called BL Heli and was created by Steffen Skaug for Silicon Lab based microcontrollers.

Both platforms are open source (source code is documented and freely available on the Web) and can be reprogrammed through a bootloader (code in the chip that directs the initialization to a programmable section in memory). There are several graphical programmers that can load/reload code to a brushless motor drive. This is important to wheeled drive motors because we require most aircraft drives to be reprogrammed to work with either RC controller-based center stick or car style forward and reverse calibration. Most importantly, the low speed control characteristics provide the same fine control capable with brushed motors.

Figure 1 is an example of three USB-based programmers that can load the firmware into compatible ESCs through the receiver cable or special socket programmers.

Now that there are ESCs with the firmware necessary to operate a brushless motor, we need to determine how to get the high speed and low torque (rotation force) of the motor to the necessary low speed and high torque to the wheels. For this, we will need a gearbox. Once again, we can leverage aircraft motors as they are often inexpensive and designed to work with the drives noted above.

For small combat robotics, the 12 mm diameter by 30 mm long motor —

also known as a 1230 motor — packs 30 watts of continuous power. For reference, a common 13 mm Maxon brushed motor is rated for only three watts continuous power. **Figure 2** is an example of a 1230 brushless motor with a 4,800 KV wind.


Figure 3.


Figure 4.

The 1230 motor has a very small mounting face pattern that appears to be designed for a single stage plastic spur gear. A metal planetary gearbox with several stages of reduction would provide a better solution. The 13 mm Maxon gearbox would be perfect as there is a large number of used 13 mm Maxon gearmotors on eBay; the gearboxes rarely have any wear, but some customization is required.


Figure 5.

Both the 13 mm Maxon brushed motor and brushless 1230 motor share the same 1.5 mm diameter shaft. You just need to core out the Maxon motor and use the motor frame as a mount for the brushless motor. **Figure 3** shows the thread mount the motor uses to interface to the gearbox.


Figure 6.

To remove the pinion from the motor shaft and allow the motor shaft and rotor to be removed from the motor, a small torch or soldering iron can be used to loosen permanent Loctite. This is shown in **Figure 4**.


Figure 7.

Figure 5 provides a sum of all the motor components. To continue the process of removing the rotor, the rear brush assembly and cover can be removed by inserting a small nail or peg into the cover hole and forcibly breaking it out. Once the brushes are out of the way, the rotor can be removed by using a small peg to hammer the pinion side of the shaft down and through the motor.

In **Figure 6**, use the same peg enlisted to push out the shaft and rotor to push out the center magnet assembly. The magnet assembly should come out leaving only the motor can and front thread mount as shown in **Figure 7**.

The inside diameter of the motor can is slightly less than 12 mm in diameter, so it must be opened up with a 12 mm drill bit. This can be done in a metal lathe or even a drill press. It is preferable to grip only the thick thread mount body area. A drill collet could be used to apply equal force in holding the motor can.


Figure 8.

Figure 8 demonstrates a lathe setup.

Finally, after test-fitting the brushless motor into the Maxon motor body, use permanent Loctite to secure the brushless motor into the body. Apply the Loctite to the upper circumference of the brushless motor and slide it into the Maxon motor body. Use the permanent Loctite to

Figure 9.

Figure 10.

also mount the Maxon gearbox pinion to the brushless motor shaft as in **Figure 9**.

Figure 10 is the final assembly. While the Loctite is curing, thread the motor into the gearbox joining the gear head and the motor. This effectively centers the motor in the gearbox and assures alignment. Final speed and torque will depend on the reduction of the gearbox (typically, either 17:1 or 68:1), size of the wheel, and drive voltage.

Paired with the aerobatic firmware-based brushless drive, this brushless drive solution could provide the same speed and torque of a motor 4-8 times its weight while maintaining the fine low speed control of existing brushed gearmotors. This solution should also be quite durable with proven components. **SV**

PROFILE: He Caught It, and I Don't Want to Cure It: Robot Fever

● by Matt Leach

The most recent season of BattleBots™ has been a real inspiration to many. My son, Dylan, is one of those. He couldn't wait for each episode to come on, and watching it gave him many ideas for making his own fighting robot. Once he learned about the Robot Battles event at Dragon Con in Atlanta, GA, he was ready to get building.

His robot, Caution is an update to an old robot design. It was created by using an old hacked BattleBot R/C toy and a kid's plastic hardhat. The bot was completed a few days before the competition, so he took that time to practice driving. He took to it quickly and was able to control his test targets well. By Sunday, he was ready for the competition!

As we walked through the doors of the International Ballroom at the Hyatt Regency Atlanta, Dylan was awestruck. The room was huge and filled with chairs pointing towards the stage where two Insect arenas stood: one for Antweights, and one for Beetleweights. To the right, the pits were about half full of builders already working on their bots and charging up batteries. We chose a table next to some familiar faces: Team Busted Nuts Robotics, the makers of Witch Doctor — a BattleBot that Dylan loved watching in action on the show.

Time passed, and other builders filtered into the pits. He got to meet and talk to Jim Smentowski from Team

Dylan getting ready for the Microbattles competition at Dragon Con *(photo by Matt Leach)*.

Dylan's one pound robot, Caution ***(photo courtesy of Michael Jeffries).***

Dylan in action at the competition ***(photo by Matt Leach).***

Nightmare about his Antweight, Micro Nightmare. Dylan was also excited to see his favorite team from BattleBots arrive: Team JACD, the builders of Over-Haul. He waited impatiently to meet Charles Guan and Jamison Go, and was so happy once he got to do so. He is such a big fan of theirs that he decided to be "the driver of Over-Haul" for Halloween this year.

We walked around the pits, met the other competitors, and checked out the other bots in the competition. Dylan asked lots of questions and was genuinely interested in everything everyone was saying. All the competitors were friendly and happy to talk to him. He took pictures of some of his favorite bots. He felt like part of the robot fighting community already.

Dylan waited excitedly for his chance to fight in the Antweight arena, and his time finally came. We made our way to the stage where his opponent — a two-wheeled titanium wedge named Tiny — waited. Prior to the match, the competitors were announced on stage to the attending audience. Kelly Lockhart, the announcer, asked Dylan how old he was. "Seven," was his response, and the audience cheered loudly. He was the youngest competitor in the competition.

After the introductions were complete, the match was to begin. Kelly said into the microphone, "3, 2, 1," and the audience yelled, "FIGHT!" The music started, and Dylan drove Caution straight towards Tiny. They bumped and pushed each other around, and Caution even got under Tiny from behind a few times. The match almost lasted the full two minutes while the miniature gladiators collided repeatedly, but Caution eventually ended up in one of the arena push-outs and lost. Tiny's driver, Clay Steere was impressed with Dylan's driving and gave him a high five, thanking him for a good match. As we made our way back to the pits, audience members and other builders were congratulating him on a great fight. Even though he lost, he was proud of how well he did.

Since it was a single-elimination tournament, we were done. Dylan took the time to talk to more people in the pits and take a few more pictures. He also enjoyed watching the rest of the matches on the large projection screens. He especially liked the matches where the spinners fought, so for his next bot he wants to build a horizontal spinner like one of his favorites: DDT, built by Jamison Go.

His ultimate goal is to be on the TV show BattleBots, but we'll take it one day at a time for now. From what he's been told, though, he will be a worthy competitor in the future. Another generation of robot combat builders is emerging, and Dylan is glad to be a part of it. **SV**

BattleBots™ Gets Tombstoned

● by Chris & Tiffany Olin

When entering the BattleBox, if you see the steely eyed gaze and confident smile of Ray Billings staring back at you, you know you're in trouble. If Ray is standing over the 250 lb harbinger of destruction known as Tombstone, you know you are in a whole lot of trouble. Ray, his son Justin, and teammate Rick Russ have been breaking bots and shattering dreams for 14 years. Together they are Team Hardcore Robotics and they have been weaving a path of destruction through such notable robot combat events as BattleBots (original series), RoboGames, ComBots Cup, Steel Conflict, Motorama, Robot Revolution, NPC Charity Open, Robojoust, and West Coast Robotics.

Now they have added BattleBots 2015 to that list with their new 250 lb heavyweight, Tombstone to battle for honor and glory in the BattleBox.

After the airing of this year's amazing BattleBots tournament, I had a chance to talk with Ray about his experience.

Olin: Ray, tell us a little bit about yourself and how you got started in competitive robotics.

Billings: I currently work as an Engineering Technician at Intel Corporation at the Folsom site. Interestingly, I started working here after I had been building combat robots for several years, and one of the key reasons I got the job was the experience from robots. The job doesn't help with the bots, but the bots helped a lot with the job!

I used to work at an adult college as the network administrator and taught computer networking. My boss at the time was a fan of the original BattleBots show, and used to bring in the recorded matches; we would watch them between classes. We decided to build something together, but unfortunately he had other things get in way and wasn't able to continue. But the idea was firmly planted, so I built something for BattleBots 4.0. I've been competing ever since.

tombstoned:

verb, informal

tomb•stoned \ˈtüm-ˌstōnd\

Utterly defeated quickly and decisively

Olin: What other tournaments have you won or placed in?


Entering the arena: Rick Russ, Justin Billings, and Ray Billings with Tombstone.

Billings: My main robot over the last decade has been the heavyweight, Last Rites. This has competed at the RoboGames and ComBots Cup events, placing highly or winning in every event since 2007. In fact, the dominance of a few select robots over this period ended up with the term "The Big Three," which were my robot, Last Rites, Gary Gin's robot, Original Sin, and Matt Maxham's robot, Sewer Snake. For every RoboGames or ComBots Cup through that period up to today, at least two out of those three — and usually all three of them — were the top three bots at each event.

It seems I am doomed to never actually win the HW class at RoboGames (2nd or 3rd each time!), but I have won the ComBots Cup twice, and am the current reigning ComBots champion.

Olin: What can you tell us about your team members and what skills they bring with them?

Billings: This started out as a father and son project with me and my son, Justin. There are many people who have been part of the team at various junctures over the years, but even today it remains very much a father/son focused team. Current team members are myself, Justin Billings, and Rick Russ. Rick used to run his own robot team, but decided he wanted a more pit-oriented role rather than driver or designer, and has been with the team for several years now.

I pretty much do all the design and major fabrication. I'm also the guy who ends up paying for most of it. Justin has turned into a pretty good driver over the years, as well as being able to handle most of the

maintenance needs at an event. He's also a big strong guy, which ends up very useful if we're taking a couple of heavyweights to an event. Rick owns his own sheet metal fabrication shop, so he has some tooling and fabrication skills that he brings to the team. Rick also handles much of the maintenance and repairs at an event.

Olin: Can you tell us a little bit about your sponsors and any material support they gave you for building Tombstone?

Billings: Team Hardcore Robotics has a long standing sponsor arrangement with NPC Robotics. NPC has provided me with drive motors and components and various engineering services for several years. They have played a big part in our success, helping us to win the ComBots Cup (twice!) and place highly at every RoboGames event. Great products from a great company!

Our newest sponsor for BattleBots is Innovation First International — the parent company for VEX robotics. IFI has created some of the most commonly used electronic speed controllers in robotics today. Three of the top four bots in this year's BattleBots tournament were running IFI products, and most top teams rely on their electronic components. I have relied on IFI speed controllers for every single combat robot I have built.

Tombstone vs. Bronco.

Olin: How do you build your robots? What tools and processes do you use?

Billings: This depends a lot on the weight class. In the larger classes, I tend to use welded 4130 steel tubing for the frame, with armor either bolted on (for non-contact areas, say the top and bottom panels) or welded on (something where contact is likely). Welded steel tubing is really pretty strong for its weight, and makes it relatively easy to address major damage at an event. I'll take some extra tubing and a welder, and we can pretty much fix anything that happens on-site, even major problems.

In the smaller classes, welded steel just ends up too heavy to work with, so I would end up using machined aluminum framing. This isn't as easy to address at an event, unless you simply make a lot of spare parts to change out in case they are needed. But it tends to make more sense from a weight management point of view for the smaller classes.

Olin: What can you tell us about Tombstone's weapon system?

Billings: The weapon motor is an Etek-R and is geared for about 2,800 RPM, which pulls about 1,000 amps at spin-up. Obviously once spun up, it is drawing much less, but not something I have measured. The weapon bar's weight is between 65 and 75 pounds.

Olin: Is there anything new or innovative on Tombstone?

Billings: Most of the different systems used on Tombstone were basically identical to my previous HW Last Rites. But the frame for Tombstone was all new, to better take advantage of the heavier 250 lb category. The frame for Tombstone was almost 20 lbs heavier than for Last Rites. Also, with the active arena for BattleBots I had to plan for more to deal with than just the opponent.

With the killsaws and hammers, I had to armor up the top and bottom to handle this. I added a 0.1 inch titanium panel top and bottom, which should have been enough to deal with a killsaw ride or two. Luckily, I didn't have to actually test how well it would have worked.

Olin: What was it like walking through the tunnel into the arena for the first time?

Billings: Even though we've done this sort of thing before, it was a little imposing. For us though, our first match was supposed to be against Beta. That team was having troubles, as some of their custom electronics were lost by the airline on the trip there, and they were scrambling to get it running.

So, we loaded and unloaded into and back out of the arena several times while they tried to get it working. Eventually, they just had to give up and we moved on to one of the alternates; in this case, Counter Revolution. We had done it so often I just wanted to get in the arena and actually have a fight!


Tombstone's battery packs after meltdown during fight with Biteforce.

In the preliminary round of BattleBots 2015, Ray and Tombstone faced RoboGames veteran, Counter Revolution. Tombstone's first hit crippled Counter Revolution's drive; the second hit disabled Counter Revolution's weapons; and the third hit splattered Counter Revolution across the arena. Counter Revolution got tombstoned!

Ray's next opponent (victim) was British import, Radioactive. Ray and Tombstone toyed with their prey before crushing him, then while Radioactive was being counted out Tombstone batted the loose pieces of Radioactive around the arena with ballistic force.

Next, Tombstone faced Witch Doctor (see the August 2015 issue's interview with Andrea Suarez). Witch Doctor took several hard hits and dealt a few of her own before the climatic hit that sent both bots tumbling. Witch Doctor ended up inverted and immobile, but in the process broke Tombstone's blade. Witch Doctor was counted out for a TKO.

Olin: What can you tell us about the damage you took during the Witch Doctor battle?

Billings: The blade that broke was made from S7 tool steel. This particular one was 1.5 inches thick, and I picked it in particular as the least likely to break. It broke through one of the lightening holes, and probably had some stress fractures prior to this match. It has seen much stronger hits in the past and survived. I didn't think it would break — part of why I chose it in the first place. But that particular weapon bar had a lot of fights on it, and they do break eventually.

Olin: How did this affect your strategy for the rest of the tournament?

Billings: It didn't affect anything really. The bar I chose for the last two matches is the one I would have chosen even if this bar had not broken.

The Semi-Finals

Next, in the semi-finals, Ray and Tombstone faced Inertia Lab's pneumatic flipper power house, Bronco. For this match, Bronco used an extra-long flipper arm and longer front forks for better leverage. At the start of the match, Bronco backed into a corner and waited for Tombstone to attack. Ray quickly obliged.

The two bots collided weapon against weapon, sending both flying. Bronco attempted a second flip but missed. Tombstone counter-attacked, scoring several big hits.

The last hit crippled Bronco's drive and sent Tombstone tumbling, causing its top cover to come off and its batteries to spill out. Bronco was counted out and Tombstone moved on to the finals.


Ray Billings (L) holding the "Most Destructive Robot" award, with Ian lewis and Simon Scott from Team Warhead holding the "Best Designer" award.

Olin: How did Bronco's special extra-long flipper affect your opening strategy?

Billings: Bronco's flipper shape was pretty much exactly what I was expecting. The only part of that fight that didn't go the way I wanted was I assumed they were going to

box rush me, and that we would be fighting out in the middle of the arena, not in their corner.

Olin: Did the damage from this match affect your next match?

Billings: We had all the damage repaired, which was really fairly minimal. I usually have several sets of batteries. I used different batteries in the Bronco match and the Biteforce match.

The Finals

In the finals, Ray's Tombstone faced Paul Ventimiglia's Biteforce — a bot with magnetic tank treads giving him extra pushing power, lifter forks on the front, and a heavy steel wedge on the back. The match started with both bots attacking aggressively. Biteforce's wedge took hit after hit from Tombstone's big blade. Just as it seemed the wedge was starting to fail, white smoke started pouring out of Tombstone, and Tombstone's big blade stopped. Biteforce dominated the rest of the match and won a judge's decision and the championship.

Olin: What was your strategy for the Biteforce match?

Billings: There wasn't any wild strategy going on in either camp to be honest. Just me trying to hit him any place other than the wedge, and him keeping the wedge pointed at me. I just ended up with an internal short in one of the battery packs which took out the rest of them.

When you are drawing the amount of amperage I do, this is always a possibility. Paul did what he had to do, and I didn't. The match would have gone much differently without the battery failure.

Olin: How was this BattleBots tournament different from past ones?

Billings: For this event, there was simply fewer bots involved. The BattleBots events in the past were epically large — 500+ big bots all in the pits — whereas this time we had less than 30. But the action was awesome, and the box was nicer than ever before. The production values of the TV part were much better this time too.

Olin: What was your most memorial moment for this event?

Billings: I think my most memorable moment was when I won the award for the "Most Destructive Robot." I was at a loss for words, which doesn't happen often for me. SV

Tips from the Pros

● by Chris & Tiffany Olin

We asked some of the BattleBots' top builders what advice and tips they would give to aspiring bot builders. Here are their answers.

Ray Billings from Team Hardcore Robotics, builders of Tombstone:

Start small. There are combat events all over the country and the world. My first combat machine was a 120 lb middleweight, and I wish I had someone tell me to start out smaller back then and move my way up. You learn so many skills and go through lots of design phases much cheaper and easier with the smaller weight classes. By far, the best way to learn is to go to an event and interact with the builders. Find an event that is local to you, go there, and watch and hang out/ask questions/get ideas. That is much more informative and inspirational than any other resource available.

Dan Chatterton from Team Wrecks, builders of Wrecks:

Find the Combat Robotics Facebook group and ask lots of questions! You will always get answers. Also, see if any builders live near you and would show you some of their bots up close or help with your design.

Figure out what your main attack will be. Then, design the body around that item and add armor where needed. Titanium is great, but steel alloys and thick aluminum fit a budget easier while still making good armor.

Ground clearance! Be sure that your bot can drive over the uneven arena floor. Many bots have lost from getting hung up.

Team Hardcore Robotics.

Team Wrecks.

Busted Nuts Robotics.

Team Raptor.

Inertia Labs.

Information Resources:

http://sparc.tools
SPARC (Standardized Procedures of the Advancement of Robotic Combat); forums and resource center.

www.buildersdb.com
Robot Builders Database; find your local robot combat events here.

Facebook Groups
www.facebook.com/groups/RobotCombat
www.facebook.com/groups/RoboCommunity
www.facebook.com/groups/NorthEastRoboticsClub

Andrea Suarez from Busted Nuts Robotics, builders of Witch Doctor and Shaman:

Start with something simple. When you see BattleBots on TV, you're watching the result of a decade or more of experience. Witch Doctor is not my first robot. My first robot was Kerminator, and it was a middleweight (120 lb) with a 10 inch saw blade from our local hardware store.

It didn't stand a chance of winning an event, but I learned how to solder, weld, wire, use a machine shop, drive, and compete. In combat robotics, "the hard way" is the best way to learn — it's just as important to figure out what doesn't work!

Go to every competition you can. Even if you don't have a robot yet, show up and talk to the builders. You'll find that this crowd is eager to help new builders get involved. This incredible community is one of the best parts of combat robotics.

Chuck Pitzer from Team Raptor, builders of Ghost Raptor:

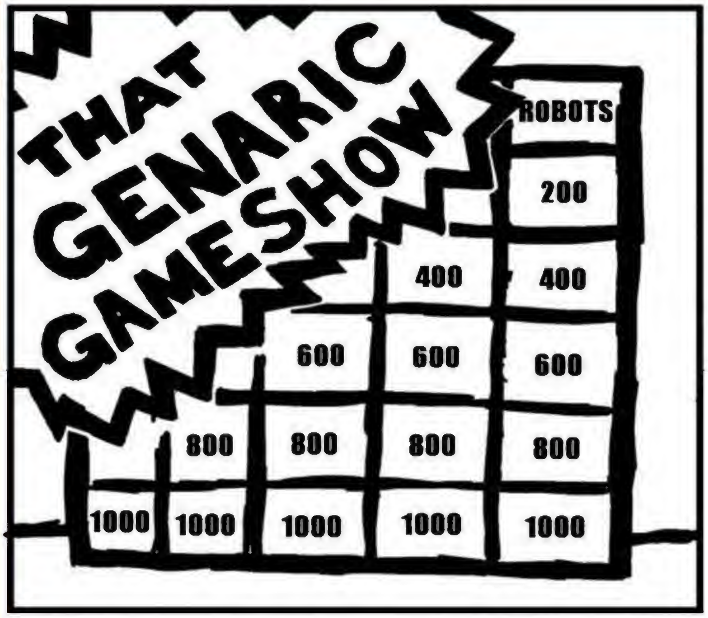
Armor up! I've seen so many robots that have a great concept and even great engineering get totally dismantled because the builder didn't consider the greatest threat in the arena: kinetic energy.

Design for speed. Slow robots are targets. Even if slightly uncontrollable, hitting a robot that moves fast is like trying to grab a fish without fishing gloves on. Speed gets you out of a lot of situations that would lose a match otherwise.

Start small and work your way up. Starting with a 220 lb or 250 lb bot means that you could spend $10K plus and end up with something that doesn't work and you never use again. Build upon what you already know. There are so many unknowns in design engineering that you are best advised to build upon what works. If you take risks (like building an articulated spinning head), build upon something that you know works well. Struggling with an untested drive system AND a risky new main weapon is a shortcut to failure.

Alexander Rose from Inertia Labs, builder of Bronco:

Build early, build often. Test, test, test. SV



MAXROBOTECH™

BY: AARON & ROBERT GROSS

GAME SHOW AI AND ROBOTICS

JJ, BOB, and Albert have never been on the game show Jeopardy!, but Watson has. Watson is a cognitive artificial intelligence (AI) developed by IBM, originally to play Jeopardy!.

For Jeopardy!, the room sized Watson memorized and understood 200 million pages of content, including Wikipedia. Like humans, Watson understood and answered questions. In 2011, the best human players lost and Watson won.

IBM continued Watson's development and cognitive training. It's now 24 times faster. It's 90% smaller. It understands all types of data. For example, it can watch and understand videos. It isn't programmed, instead it learns. It has easy web based cloud access, where tens of thousands of developers are working applications.

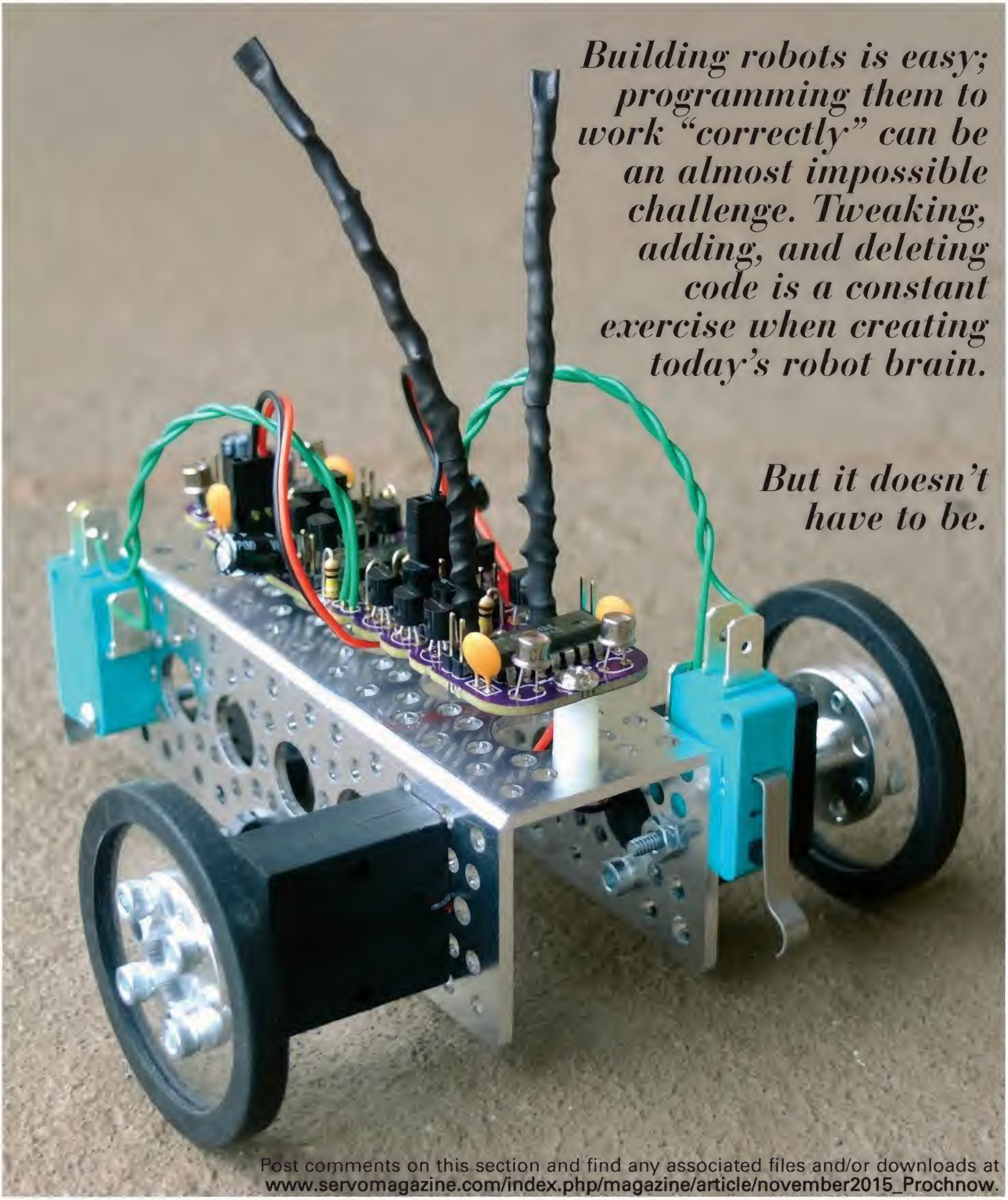
A fun and useful demonstration of Watson is the TED Talks TV show search engine. Here Watson provides answers to questions and recommends shows relevant to your question. This can be found here watson.ted.com

Watson is also helping the small humanoid robot Nao from Aldebaran. Using Watson, Nao can speak with realistic intonation and use appropriate hand gestures, and even show impatience and sarcasm.

AI has come a long way. Providing better search and robots showing emotions are just the beginning.

Microcerebrum

By Dave Prochnow



Building robots is easy; programming them to work "correctly" can be an almost impossible challenge. Tweaking, adding, and deleting code is a constant exercise when creating today's robot brain.

But it doesn't have to be.

Post comments on this section and find any associated files and/or downloads at **www.servomagazine.com/index.php/magazine/article/november2015_Prochnow.**

Build a Better Bot Brain

There is a different path for creating robust powerful robot brains. However, you're gonna have to go backwards to go forward with these alternate cerebrums.

If you ask 10 robot builders to describe the brain used for controlling h/is/er creation, you'd hear names like Arduino, PICAXE, and BASIC Stamp. Maybe even an Intel® Edison, LEGO®, and Raspberry Pi will make the top 10 in your robot builder's survey. Regardless of the brand, all of these "brains" are, in fact, microcontrollers which act like miniature programmable computers for driving motors, reading sensors, and flashing light emitting diodes (LEDs), and, in general, controlling all of a robot's hardware.

The emphasis on "all" is important in the above microcontroller generality: Without programming, these computerized robots are just expensive doorstops. Programming a microcontroller is such a vital, yet daunting task that the time consumed by developing robot software can sometimes dwarf the time spent building the hardware.

Beyond the "resistor and transistor" robots of yesteryear, today's program-less robot brain is built from digital integrated circuits (ICs) that require no programming. Just design, build, and go. Likewise, these aren't simple robots, either. These are complex sensing/reacting robots that are able to provide robust functions without high cost components or labor-intensive programming. Make no mistake — even within the context of the best ever written generic description of robots, these microcontroller-less robots are still robots.

So, rest assured, this article is not some wild, harebrained, loopy attempt at discrediting microcontrollers for employment as robot brains. Rather, it is a proposal for an inexpensive complete robot controller alternative that can deliver the equivalent functionality of programmed control without the coding chore overhead.

So, let's begin at the beginning ... bug brains!

An Introduction to Insects

Insects rule the world! Well, more specifically, insects (or, even more specifically, Class Insecta of the Phylum Arthropoda) are the largest group of animals in the world. Representing nearly three-fourths of all known animal types, insects collectively include around one million identified species. Unfortunately, most of us humans think that all insects are pests. Swat a mosquito. Pest! Swish away a fly. Pest! Smash a cockroach. Pest! Pest! Pest! Not every insect is a pest, however.

Honey bees, butterflies, and ladybird beetles are all benevolent to humans. Some insects with disgusting habits (e.g., carrion beetles) benefit humans by scavenging on dead animals and decaying vegetation. (Hey, it's better a beetle does this dirty work than us, right?) Then, there are the countless numbers of insects that serve as a food source for vertebrates, such as caddisflies which are beneficial to freshwater fish.

Yes, it's a mixed bag with the insect world. From disease vectors and agricultural pests to pollinators and essential food chain units or producers, insects can be both a nuisance and an invaluable ingredient for life.

While all insects have six legs (see **Figure 1**), it's the size and shape of a bug's leg that determines the movers from the swimmers and the jumpers from the walkers. Regardless of the type of locomotion, an insect's leg begins at the body with the coxa segment, followed by the trochanter, femur, and tibia, then, concludes at the tip of the leg with the tarsus.

In order to move, an insect must have some sort of controller or system for triggering leg movement. This control is the function of an insect's nervous system.

The nervous system in insects consists of ganglia or masses of nerve cells that are organized in the insect's head (i.e., brain), as well as a long nerve cord that runs along the underside of its body (i.e., ventral nerve cord). Nerves also extend from these centralized ganglia into other parts of the bug's body.

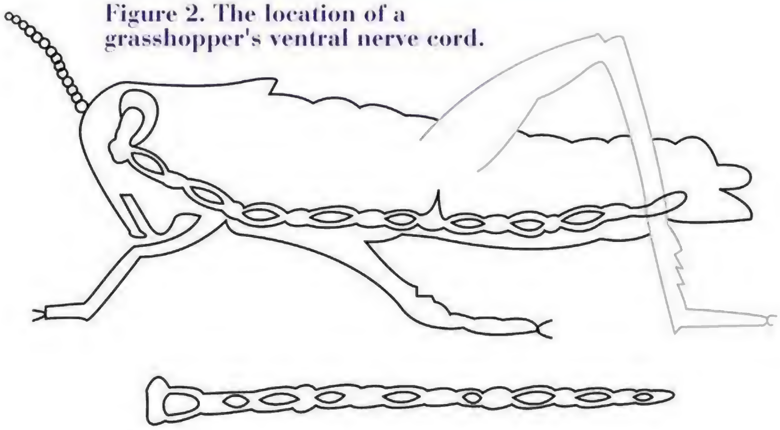
Insect Movement

Getting from point A to point B looks effortless for an insect. Although the reasoning behind a bug's movements seems equally pointless to us, they do appear to be relentless in the pursuit of their goal — whatever that goal is.

Ants, butterflies, and bees all act like they're going to some big event. But what? A lot of seemingly anxious, well orchestrated mass movement — but where's the party,

Figure 1. Newly emerged Eastern Lubber grasshoppers.

Figure 2. The location of a grasshopper's ventral nerve cord.



bugs?

Furthermore, insects have six legs (and/or one or two pairs of wings). How on earth are ants, for example, able to coordinate such rapid movement between six different appendages while climbing trees, scooting through grass, and tip-toeing along a twig?

The answer lies inside a bug's head — its brain. Or, more specifically, brains! Original neural/motor research in entomology was focused on the observable: fixed, repetitive patterns of leg movements and coordination in insects. Fast forward 80 years and research showed that walking movements were not fixed but highly adaptable.

Add another 30 years of research: Individual legs can function as independent units with walking patterns developing via coordinating neural pathways within the nervous system, sensory inputs, and mechanical attachment through the ground.

Each of these research bits can be collectively combined into a much better picture of insect movement. Therefore, taken as a whole, the explanation for the locomotion in bugs ranges from readily observable biomechanical factors through central and peripheral neurobiological controls, culminating in muscular force development. Whew! That's a lot of stuff going on inside that ant's head!

Watch the movement of some hexapod robots and you will notice that, typically, the leg movement pattern begins with the hind legs; moves onto the middle legs; and, then, concludes with the front legs on either side. This is called a simple walking gait. Regardless of the insect leg shape or function, they all operate pretty much the same. There are two excitatory motor neurons: one fast and one slow. As you might guess, the operation of these neurons follows along with their name. During a slow gait, for example, the slow neuron fires in bursts which contribute to muscle electrical potentials resulting in joint extension. Then, at faster speeds, the fast motor neurons kick in and generate some electrical spikes, which suddenly produce rapid extension of the joint, greatly increasing the joint velocity.

How are these neurons controlled? A local control system called the thoracic ganglion controls the basic insect leg/joint gait and stance patterns. In turn, the ganglion is able to adjust muscle timing and force on-the-fly (no pun intended).

An overview of the ganglion at work would consist of monitoring each joint and ensuring that they alternate between extension and flexion. This basic timing function is often performed by central pattern generation (CPG) circuits.

Insects have evolved local control circuits that work between CPGs for each leg joint and sensory reflexes. Sensory reflexes are used to adjust the strength of motor activity, thereby creating effective "burst" movement for environmental hazards (e.g., climbing over a branch). On a more general electrical level, currents through the thoracic ganglia make a leg on the "negative" side flex and a leg on the "positive" side extend.

The discovery of neural/muscular/electrical leg/joint movement potential is a rather common laboratory exercise that involves a decapitated cockroach dissected from the dorsal surface with the alimentary canal removed. The insect is suspended by hooks from its anterior and posterior ends, so that the legs are completely free. Recordings are made of the nerve and muscle action potentials. The tracheae are left intact during this dissection — to improve the condition of the ganglion.

HA! Take "that" you dirty cockroach ... oh, and, "thank you" for helping us understand insect leg movement.

In a Bug's Brain

The basic fundamental unit of the insect's central nervous system is the neuromere. This neuromere is a collection of neurons that are responsible for receiving sensory input and thereby controlling the movement of that segment. In the case of a thoracic segment, this movement would be a leg movement. Remember our poor dissected cockroach? These segment neuromeres can be combined and fused into a clump or ganglia. These ganglia, in turn, are formed into the pre-oral, dorsal brain, the subesophageal ganglion, and the thoracic and abdominal ganglia of the ventral nerve cord as shown in **Figure 2.**

The thoracic ganglia are contained in body segments which house the bug's legs. Inside the head and dorsal brain, there are three fused ganglia:

- **Protocerebrum** — the first ganglia; associated with processing visual sensory information from the optical organs.
- **Deutocerebrum** — the second ganglia; associated with processing tactile sensory information from the antennal organs.
- **Tritocerebrum** — the third ganglia; associated with integrating the information from the first two ganglia, as

well as serving as a link between the brain and ventral nerve cord (VNC).

• The final ganglion — the **suboesophageal ganglion** — monitors, regulates, and controls the sensory and motor activities of the mouth parts.

Bug brains are pretty impressive, eh? Even more so when you consider the flexibility and diversity in the number of different roles that insects play. Sure, some are pests, but even those bugs that annoy us are very resourceful and resilient in their deliberate actions to continually bug us.

Now, on to bot (bug) brains!

Microcontroller-based Robots

If you want to add some "control" (i.e., movement and brains) to your robot, you'll most likely reach for a microcontroller. Self-contained with its own programmable memory, a microcontroller is a small discrete digital device that shares a lot of characteristics with a desktop PC.

Some robot builders, however, might argue that one of the biggest drawbacks with using a microcontroller on a robot is that you must write your own software program for driving your creation. This is not an insurmountable obstacle. Merely another piece of the puzzle for turning a pile of parts into a functioning robot.

Ironically, programming most microcontrollers requires a PC. Go figure, eh? Luckily, the big names in robot microcontrollers (e.g., Arduino, Stamp, PICAXE, RaspPi, etc.) have easy-to-use PC-based programming environments (i.e., Integrated Development Environment or IDE) that can be coded in Basic, C, or C++ computer languages. So, the entire robot microcontroller programming process sorta goes like this:

Develop program on PC -> Compile program -> Load program -> Test on robot -> Repeat process.

No matter how much you think you know about programming, you are bound to make a coding mistake. Some mistakes — also known as "bugs" (an ironic term when you consider the discussion earlier in this article) — are relatively easy to find and correct. Others can be downright inscrutable in their nature and seemingly impossible to eliminate.

If you don't like being saddled with programming and its inherent debugging tasks, that's okay. Either hire a programmer or find another brain for your robot. A brain that doesn't require any programming — like my Microcerebrum that is outlined in **Figure 3.**

Introducing a Bug-Inspired Bot Brain

Just like Mark W. Tilden's BEAM robots (i.e., Biology, Electronics, Aesthetics, Mechanics), not all robots run on microcontrollers. Remember, Mark championed a non-programmed robot building movement that grew with a life of its own to become one of the elemental building blocks in every budding robot builder's vocabulary.

This was a true 'less is more' convention that relied on digital electronics coupled with discrete components, but

Parts List

Qty	Description
9	Two-pin headers
4	Three-pin headers
20	2.2M ohm resistors
10	100K ohm resistors
4	2N3906 transistors
10	2N3904 transistors
3	CD4011 ICs
4	CL905HL photocells (for vision input)
4	.0047 μF capacitors
2	100 μF capacitors
1	.1 μF electrolytic capacitor
1	10 μF electrolytic capacitor
1	1N4001 diode
1	3 mm Red LED+resistor
1	3 mm Green LED+resistor
2	SPDT Snap-action simulated roller switches (BG Micro part #SWT1116; for tactile input)

shied away from all microcontroller support.

Remarkably, BEAM robots are not brainless, in spite of their lack of coded behavior.

Recalling our quick review of the insect's central nervous system earlier, we can see that focusing on the neuromeres or segment ganglia might be a great candidate for guiding us in the design of a robot brain. These segmented or metameric ganglia are ideal for mimicking in a code-less controller circuit.

'Hold on there,' you might say. Isn't this the domain of BEAM robotics? Well, yes and no. Yes, BEAM designs tackle many of the sensory and motor I/O behaviors that are equivalent to a grasshopper's neuromere, for example. But, no, a BEAM circuit isn't really suited for (nor was it designed for) handling every possible robot configuration with one circuit. For example, a solar-powered BEAM bot would be pretty useless in the dark.

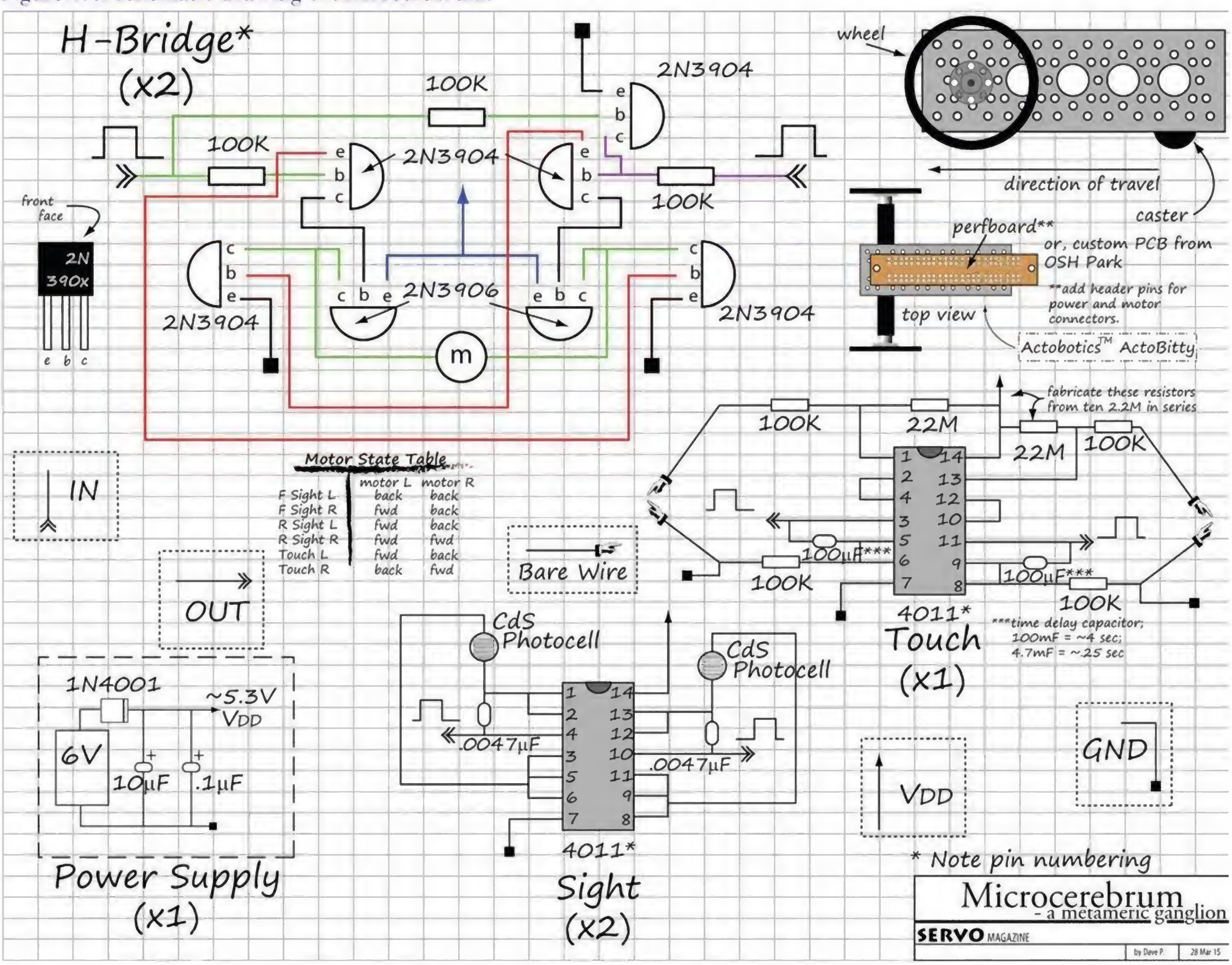
Rather, we can try to emulate an insect's metameric ganglia with a handful of discrete electronic components along with, maybe, a digital IC or three.

No microcontroller is needed. Therefore, no coding will be necessary for this robot brain. Also, remember this is not a BEAM circuit nor is it intended to be a BEAM replacement. It is merely a robot brain alternative to using a microcontroller. And I call it Microcerebrum.

At the very heart of building this better bot brain is CMOS. Complementary Metal-Oxide-Silicon or CMOS ICs are thin sandwiches of positive (PMOS) and negative (NMOS) channel MOS transistors that consume small amounts of power from modest (i.e., 3V to 12V) power supplies. By the way, low power consumption is an excellent trait for any robot brain.

Furthermore, a CMOS IC plays well with discrete components, as well as TTL/LS ICs (Transistor-Transistor Logic/Low-Power Schottky; that's the same kind of IC — 74HC240 — used in Mark's Bicore BEAM robots). Likewise, CMOS ICs come in a dazzling array of configurations and logic patterns. Those two attributes are perfect for trying to

Figure 3. A schematic drawing of Microcerebrum.



emulate a metameric ganglion.

Three possible CMOS chips that could handle our digital neuromere copycat are:

- **CD4049** — Hex Inverting Buffer — six inverting (e.g., IN low, OUT high) buffers.
- **CD4011** — Quad NAND Gate — four NAND (e.g., IN low + low; OUT high) gates.
- **CD4066** — Quad Bilateral Switch — four analog switches controlled by pins (e.g., pin to VDD, closed).

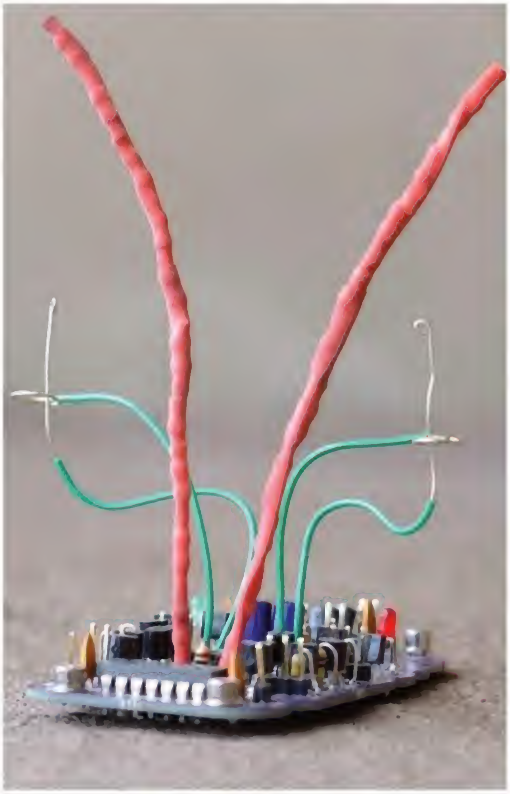


Figure 6. Testing the proof-of-concept board.

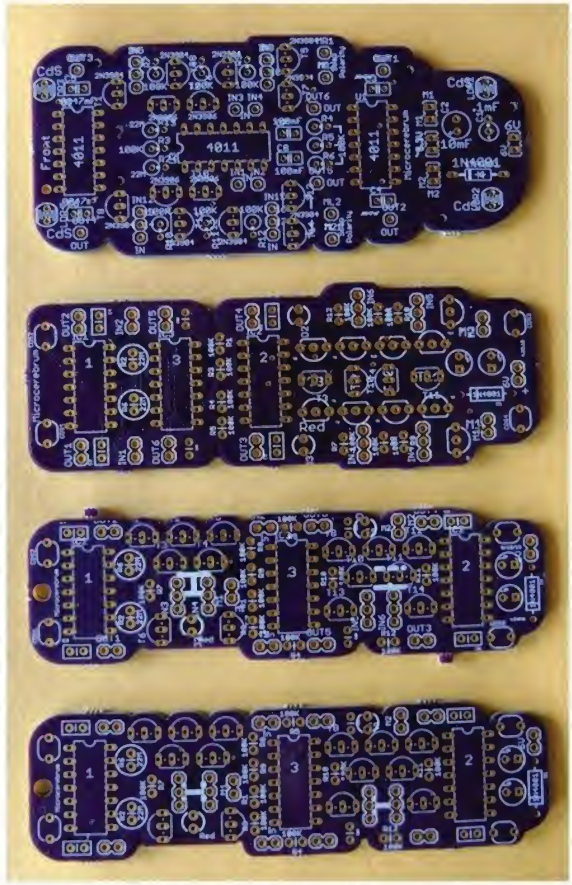


Figure 4. The evolution of Microcerebrum — from the "proof-of-concept" at the top, to version 1.0 at the bottom.

While each of these CMOS ICs are very reasonable in cost, when the number of support components for each chip is factored in, the ideal candidate is the CD4011 quad NAND gate. Remarkably, the CD4011 IC is also able to handle all of our sensory input requirements, as well as play nicely with a gutsy seven-transistor motor H-bridge (a circuit that is based on a Mark Tilden design).

In spite of this easy IC choice, it still took several iterations of the Microcerebrum design to get everything right (see **Figure 4**). Additionally, even though this is supposed to be a robot brain, the final version 1.0 printed circuit board in **Figure 5** includes an image of a grasshopper on its bottom layer.

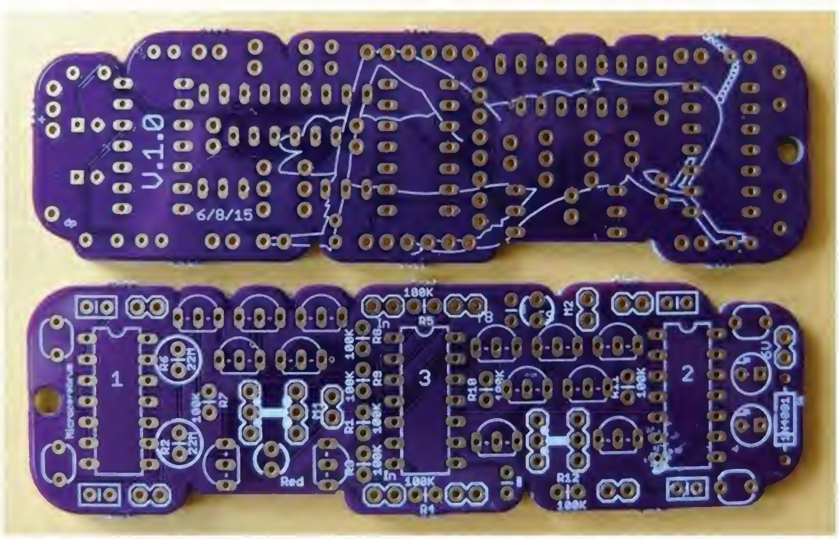


Figure 5. Two sides to Microcerebrum: left brain versus right brain?

A lot of testing was performed on the "proof-of-concept" board (see **Figure 6**). During this testing, a major change was made to the connections between the CD4011 ICs and the H-bridge outputs. In order to maintain a reasonable amount of flexibility with Microcerebrum for controlling a robot, header pins are used for every CD4011 output. Therefore, female-to-female jumper wires can be used for altering the behavior of these outputs (see **Figure 7**). In the final tally, there are six sensory inputs that are channeled through the three CD4011 ICs (four visual inputs and two tactile inputs); the resulting outputs (as shown in **Figure 8**) are open for inserting jumper wires to the two H-bridge motor drivers.

As you look through the **Parts List** for Microcerebrum, there are three items that stand out as oddball components that not every robot builder might have laying around the workbench. Well, actually, two of the items are very similar — the red and green LEDs. These

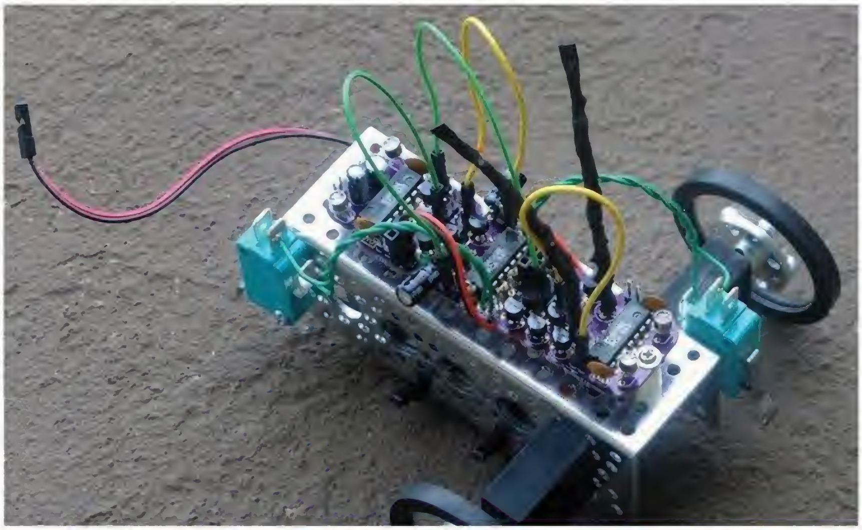


Figure 7. Jumper wires are used for controlling the behavior of Microcerebrum.

are not your "normal" 3 mm LEDs. Due to the limited board space available on Microcerebrum, standard LED and resistor combinations were eliminated in favor of Kingbright all-in-one LED + resistor packages. That's right; a current-limiting resistor is already included inside the LED. These are Mouser part numbers 604-WP710A10ID5V (red) and 604-WP710A10SGD5V (green).

The final oddball item is/are the photocell(s). I know this is going to sound crazy, but I tested 30-40 different photocells for enabling/disabling the CD4011 output gate, and the only ones that worked were surplus ones manufactured by Clairex and available from BG Micro (e.g., CL905HL).

Yes, you can use other variable resistance controls (e.g., trimmer pots), but if you want a photosensitive robot vision ganglion, then you need to use BG Micro part number RES1439.

While we're looking at the **Parts List**, you will also notice that 20 2.2M ohm resistors are specified. You're right, that circuit board can't hold 20 resistors! Rather, two extra long 22M ohm resistors are fabricated from these individual resistors. By soldering 10 2.2M ohm resistors together serially, we are able to concoct a 22M ohm resistor (see **Figure 9**).

A small strand of 30 gauge wire-wrap wire is then soldered to one end of this long resistor and run down to the other end. This combination is then secured inside some heat shrink tubing and soldered onto the board. The final eccentric look of these long resistors is used for imitating an insect's antenna. If that mimicry is too goofy for your taste, a standard 22M ohm resistor can be substituted in the circuit.

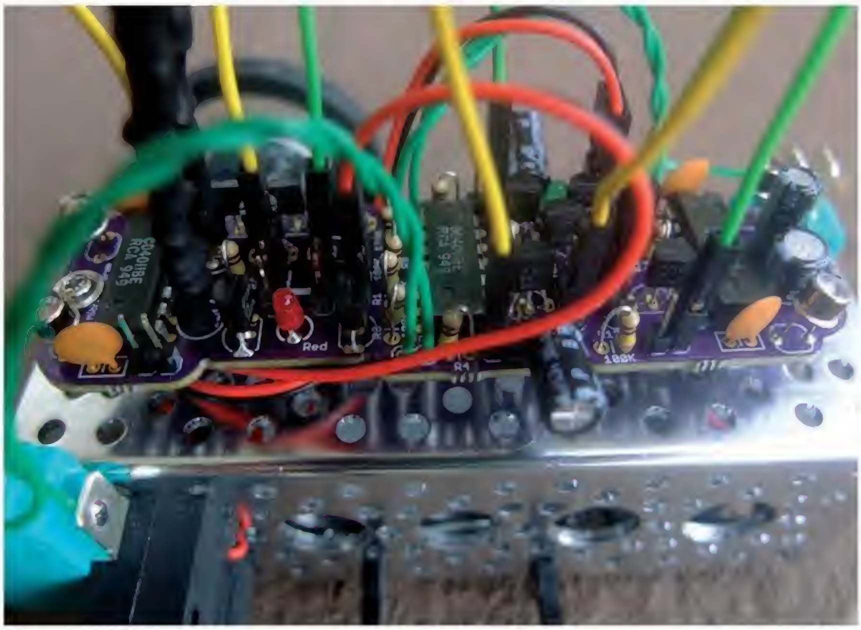


Figure 8. There are six sensory inputs and two movement outputs on Microcerebrum — similar to a fused metameric ganglion on the ventral nerve cord of a grasshopper.

I included a mounting hole in Microcerebrum for attaching it to an Actobotics ActoBitty robot (see the July 2015 issue). This inclusion enabled me to raise the board in **Figure 10** above the aluminum chassis of the robot, preventing short circuits.

Oh, and remember those headers for the CD4011 outputs? This modular design feature also allows you to selectively use only the sections of the board that you need for enabling a programming-less robot brain on smaller projects (see **Figure 11**).

Finally, I put my bug to work. By mounting a

Figure 9. Each 22M ohm resistor is fabricated from 10 serial 2.2M ohm resistors soldered together.

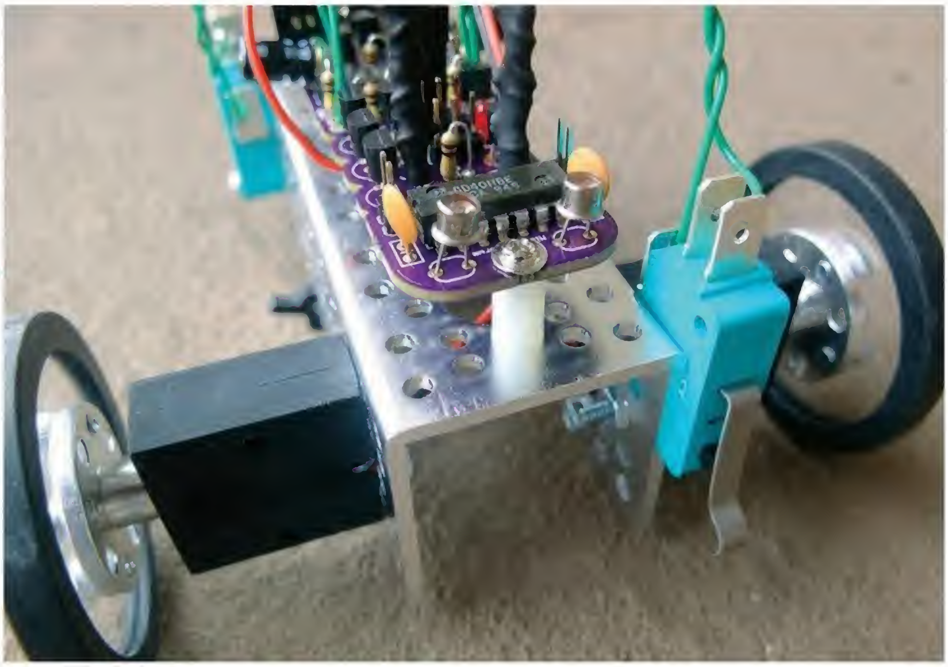


Figure 10. Board is raised off the aluminum ActoBitty chassis with a nylon spacer.

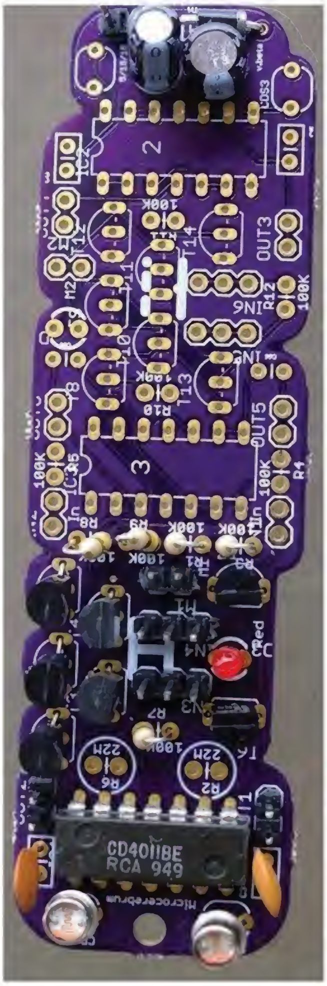


Figure 11. Select portions of Microcerebrum can be used for powering smaller projects.

Microcerebrum on an ActoBitty and adding a scavenged solar-powered yard light to its back, I created my own ActoBuggy! By tweaking the inputs and outputs with jumper wires, I was finally able to make my ActoBuggy drive around the yard seeking the sunniest locations.

This relocation action kept the solar-powered yard light in full sun during the day and recharged its onboard NiCd battery. Then, every day I swap out a different NiCd battery and cycle it through the ActoBuggy battery pack. Therefore, the Microcerebrum helps feed itself by keeping its batteries fully charged.

That's either a brain with a reason or a reason for a brain. Take your pick. **SV**

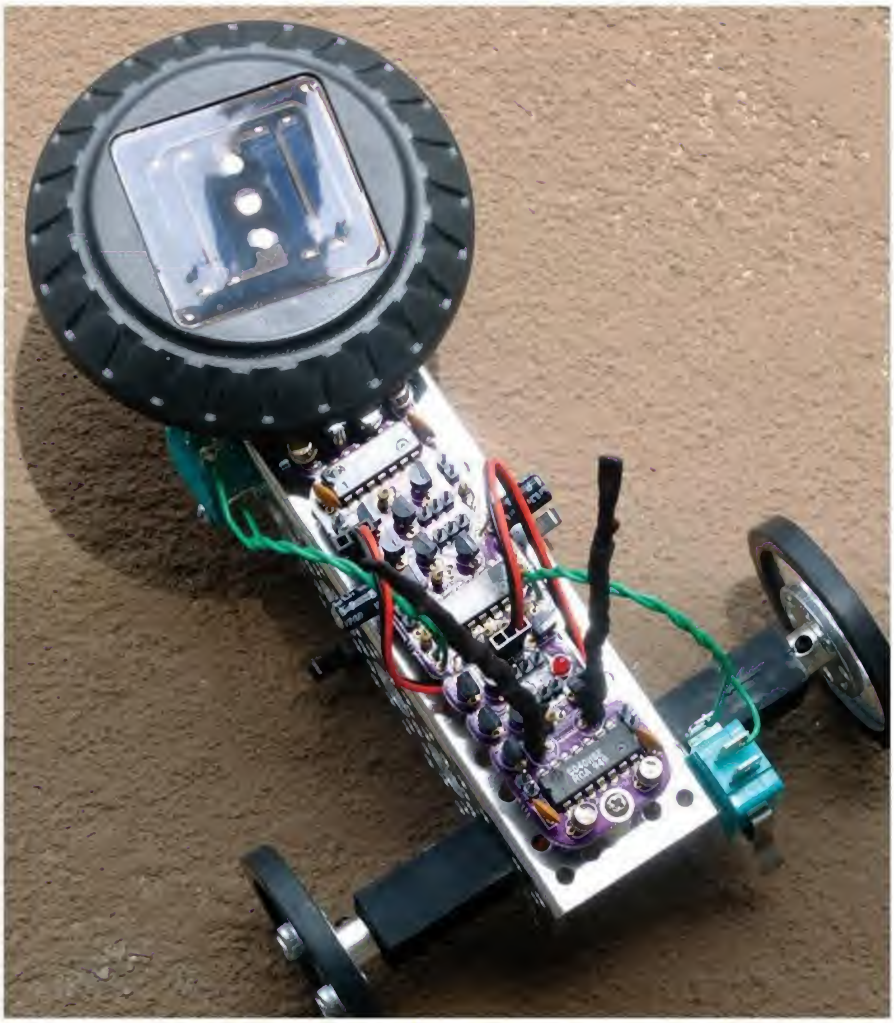


Figure 12. Putting ActoBuggy to work — charging NiCd batteries in a solar charger.

A Breath of Fresh Air

By Steve Koci

What Can It Do for Us?

When building animatronics, it is important to avoid falling into the trap of always trying to use the same mechanisms and techniques to solve every build challenge. We all have our favorite methods, but sometimes you need to implement other techniques in order to solve a unique problem.

We are often confronted with the need to rapidly move heavy loads. Servos and motors are not designed for this, so we need to look for another solution. To paraphrase Bob Dylan, the answer may be blowing in the wind.

When most people consider using air as a method of movement, the first thought that usually comes to mind is the use of pneumatics. Although this is the most popular and the one we will primarily discuss here, there are a few other "windy" options you may want to consider. We will take a quick look at a couple, and maybe they will inspire you to come up with a new concept yourself!

The use of pneumatics provides the prop builder with a unique set of qualities that are hard to achieve using any other method of movement. They are able to lift much heavier loads than servos and can move extremely fast. Once you start using pneumatics, you will find a variety of applications where they will be the optimum solution. While their use may be a bit intimidating initially, harnessing this power is no mystery.

Many feel there is a steep learning curve required to harness this elusive power of pneumatics. Do not be hesitant to give them a try as the basics are fairly straightforward. Granted, you do need to watch what you are doing in order to keep things safe. However, mastering this skill will allow you to accomplish many tasks that may have previously been impossible for you to do using other methods. Like any other technique, they are not without their own set of drawbacks. Pneumatics do require some extra equipment in the form of an air compressor, hoses, and fittings, as well as the required electronics necessary to control it all. There is also the need to deal with the loud noise of the compressor, along with the blast of air exhaust that accompanies the use of pneumatics. There are a few ways to deal with this, and we will cover those later in the article.

Pneumatics also include the limitation of being either on or off. Standard cylinders using the traditional setup we will discuss here lack the proportional control we have when using servos. There are models available that are controllable, but those are beyond the scope of this article. I will add that topic to my list of future projects.

Will I Need to Take Out a New Loan?

The $64 question always seems to be, "How much will

Post comments on this section and find any associated files and/or downloads at **www.servomagazine.com/index.php/magazine/article/november2015_Koci.**

Figure 1. A four-way, five-port solenoid.

it cost me to get started?" It may seem like an excessive amount of "stuff," but like with most new projects, you need the proper equipment to do it right.

Air Compressor

My suggestion when shopping for a compressor is to buy a larger model than you think you will need. Once you own one, you will find many other uses for it. Nail guns, staplers, and grinders ... oh my, more cool new tools to buy! I often wonder how I ever got anything built without the use of air!

Check the power requirements and be sure you have the electrical capacity to support your chosen compressor. They come in both 120V and 240V models. I run two large 120V compressors for my Halloween display and had to add two dedicated 20A breakers to run them. For me, the expense was money well spent. However, you must factor this in when deciding on the extent that you will utilize pneumatics. Make sure your service has sufficient amps and that you use appropriately sized extension cords.

Another factor to consider is the CFM, or cubic feet per minute. The higher the better — especially if you plan to use it for other work projects, like running air wrenches.

Expansion chamber mufflers and intake silencer filters are available that can help mitigate the noise created by your compressor. There are commercially made units, as well as some do-it-yourself models, so check what may be available for the unit you choose.

When shopping for a compressor, you will need to decide if you are going to go with an oiled or oilless model. The oil free units are usually less expensive, but can be considerably louder. For anything but occasional and light use, I would recommend going with a unit that uses oil. The maintenance is a bit more and the unit itself is heavier, but the added durability and quieter operation will be worth it in the long run.

New units have the benefit of coming with a warranty, but you may want to check for used models. I have seen many on Craigslist that would fit the requirements of the DIY prop builder at a substantial cost savings over a new unit.

Solenoids

A solenoid is an electrically operated valve that plays a crucial role in our system by controlling the airflow to our cylinders. By switching the power on and off, we can either extend or retract the cylinder shaft by alternating the airflow between the two outlet ports.

There are plenty of choices when shopping for solenoids. They come in a variety of sizes, with two to five ports. Port sizes also vary, so check the flow rates of the different models and choose one that fits your design requirements.

You must also decide if you will use 12 VDC, 24 VDC, or 110 VAC to power them. It is much easier to swap things around if necessary if your entire system uses the same voltage. I prefer to use 12 VDC as they are safer to have in an environment where my guests could conceivably come in contact with them.

My preference is to use four-way five-port solenoids. These are the most useful and the unneeded ports can always be plugged. When designing my props, I like to plan for the future as something I build today will one day be dismantled and the parts incorporated into a new project (**Figure 1**).

Cylinders

Here again, you have many configurations to choose from as cylinders come in different designs, mounting options, and fitting sizes. The style most commonly used when building animated characters are the cylindrical models which is what we will focus on here.

Single-acting cylinders have one air inlet that allows for the cylinder shaft to move in one direction — usually out — and then use gravity or a spring to return it to its original position.

Most of mine are double-acting cylinders as these provide me with the widest range of control when incorporating them into the body mechanisms of my characters. These have two air inlets which allow you to control the airflow in both directions (**Figure 2**).

Stroke length and bore size need to be fit to the design requirements of each build. Larger bore sizes are more powerful but require more air, so you want to select a model that will do the job but not waste air.

I prefer the universal mounting option which will let the cylinder pivot. There are times when you will need a parts movement to be restricted to a single plane of motion, however, so choose your mounts carefully.

Speed can be adjusted by the use of flow controls and regulators. I run high pressure from my compressor and then adjust the pressure to each character with a regulator. Each cylinder port is installed with flow control fittings so I can further dial in the speed at which they operate. This allows me to have a cylinder respond at one speed when it extends and a different speed when it retracts.


Figure 2. Comparing a single- and a double-acting cylinder.

Fittings

Fittings are available in either metric or standard, so it will save you a considerable additional investment if you standardize. Mixing sizes — even if they seem to fit — will cause you many headaches because it could strip threads or cause connections to leak, costing you time and money. I don't know about you, but both of those things are in short supply in my shop!

The push fittings are wonderful because it makes hooking up air lines a quick and easy process. They come in many styles and sizes, so purchase a selection of different ones so you are prepared to adapt your layout on the fly (**Figure 3**).


Figure 3. My fittings assortment will fit most needs.

Air Lines, Hoses

You can select between several different sizes, colors, and qualities of air line. After trying numerous different styles, I have settled on black polyurethane airline for its kink resistance which comes in handy when having to make tight turns on an armature. I have not been satisfied with the look or performance of cheaper alternatives, and have had it fail in spectacular fashion. Talk about a loud air exhaust!

I use 3/8 inch line to each individual figure up to the regulator, and then switch to 1/4 inch hose to each cylinder.

Regulators

Use one for every prop which will enable you to dial in the appropriate pressure for each one. We only use enough air pressure to get the job done. It is safer and you do not want to waste air as it is too valuable of a resource.

Storage Tanks

These can be an extremely useful addition to your air delivery system. I use these for inside displays or where it is impractical to run an airline from the compressor to the prop (**Figure 4**). They can be a viable solution in many cases where you have located a prop in a remote location that uses a small cylinder that doesn't get activated often.


Figure 4. An 11 gallon storage tank in its hiding place.

Figure 5. Four solenoids mounted on a manifold.

Figure 6. An example of what may be included in a complete kit.

Figure 7. Simple pop-up mechanism.

They also can provide added air storage to your system which will minimize how often your compressor needs to run. Remember, though, that when it does kick back on, it will run longer in order to refill all the additional storage tanks and lines.

Designs that require large amounts of air will benefit by having a storage tank nearby to satisfy its requirements. Active props with multiple large cylinders running for extended periods can be air hogs!

Manifolds

You can employ a manifold when you are using multiple solenoids (**Figure 5**). This allows you to use a single air line to supply several solenoids. It also combines the exhausts into two lines which will further simplify your design.

You may want to start with a simple pop-up kit available from a variety of vendors, one of which is Fright Props (see **Resources**).

A kit such as this can be utilized when constructing a simple pop-up which is a perfect introductory pneumatic project (**Figure 6** and **Figure 7**). I used something similar for my stand-up skeleton.

For the skeleton, I disassembled the spine and installed a threaded rod with some nuts and washers attached. I put this in the spine and added some Great Stuff expanding foam and reassembled the spine. Great Stuff can be an incredibly useful product, but does require a little extra care

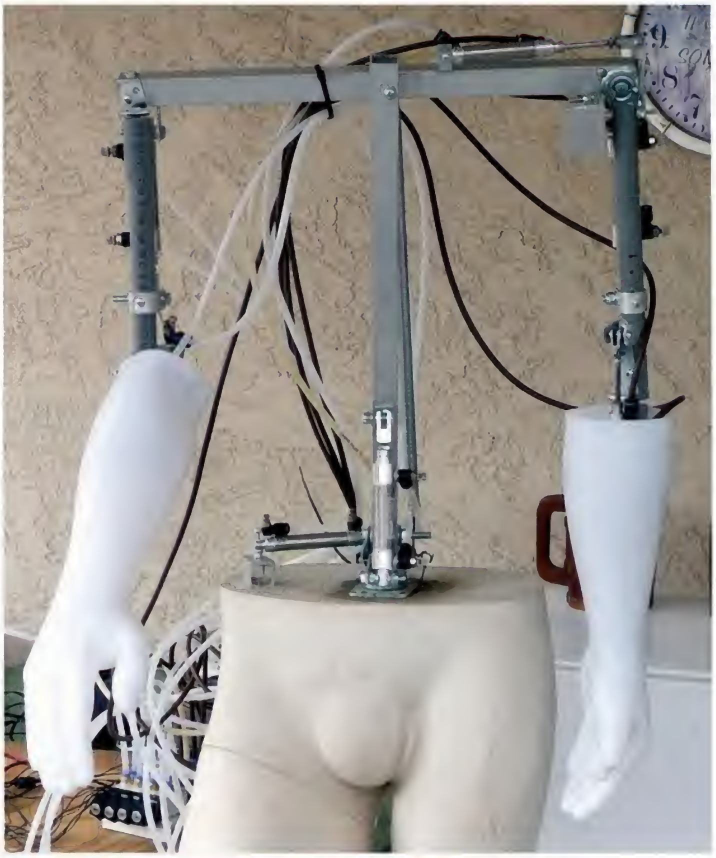


Figure 8. Pneumatic mechanism and its original installation.

Figure 9. Right shoulder assembly.

when using. Old clothes and gloves are a must as removing dried Great Stuff from skin is nearly impossible! If you're using a character that will be clothed, assembly will be much simpler.

I mounted the cylinder vertically and connected it to the threaded rod which allowed my character to rise from his seat, deliver his speech, and sit back down.

I have listed several more of my favorite suppliers in the **Resources** section.

Another source for parts is eBay. If you do your research, there can be some outstanding deals to be found (although they seem to be harder and harder to come by). I have managed to purchase an array of used components in this manner with a lot of life remaining in them.

A Zombie Comes to Life!

I originally designed the mechanism for the Big Red build (from last month) using pneumatics, but was unhappy with the restriction of only having in and out control. Although the mechanism was not the optimum solution for that project, I could easily repurpose it for my lunging zombie (**Figure 8**).

This mechanism consists of seven cylinders providing the body movements, and one to slide him in and out. I wanted to keep the design to a maximum of eight cylinders as I could then use a single programming board and two four-valve manifolds. The addition of any more cylinders would have greatly increased the complexity of the project without adding much to the performance of the mechanism.

The cylinders were attached using custom fabricated parts (**Figure 9**, **Figure 10**, and **Figure 11**). Just to be clear, these were not professionally made but cobbled

RESOURCES

Automation Direct — http://tinyurl.com/yft5tmf
Monster Guts — http://tinyurl.com/npjgcsm
Helpful Pneumatic links list — http://tinyurl.com/otjx6v4
Fright Props tutorials — http://tinyurl.com/pwcctjg
My website — www.halstaff.com
My YouTube channel — http://tinyurl.com/nma2doj

Figure 10. Left shoulder detail.

Figure 12. Fully assembled zombie.

together in my garage using bits and pieces I had on hand. It always was my plan to redo many of the components in order to improve their appearance. However, they worked fine, and other projects demanded my time so the prototypes are what I went with. Although you have the opportunity to see "under the hood," he will be clothed and my guests will never know (**Figure 12**).

So, with the torso already built, half my work was already completed.

I still needed to design and build the slide mechanism. Since it would be supporting a full torso and head that would be violently thrashing around, I chose to build it out of steel. I cut the base so that it would fit inside the tripod that supports my hanging skeletons. This would partially hide it when it wasn't activated, as well as provide some additional support to keep it in place (active pneumatic props can be a challenge to secure in place).

I modeled my slide mechanism after the one built by Canyon Trail Cemetery at **http://tinyurl.com/pxzfrj9**. I liked the simplicity of the design, as well as the ease of construction. It utilizes materials easily obtained from a big box hardware store. I chose to weld mine together, but the design could be adapted to allow you to assemble it by using bolts and lock nuts instead (**Figure 13** and **Figure 14**).

Figure 11. Base pivot and tilt mechanisms.

It's controlled by a PICAXE Octopus

board that was developed by me and Steve Bjork (**Figure 15**). This board allows me to record a sequence of movements which are stored in EEPROM memory to be replayed when triggered. The board also incorporates a stereo audio player which provides both ambient and activation audio tracks.

The new Banshee board that Steve has in development (**Figure 16**) will combine the qualities of the Octopus and Frankenstein three-axis controller, as well as his Thor thunder and lightning controller into a super controller. Watch **http://haunthackers.com** for updates on its release status.

Check my YouTube channel after Halloween for a completed video (see **Resources**). Our zombie will be dressed and painted with all the light and fog effects included. He should be quite a hit!

Figure 13. Completed slide assembly before paint.

Quiet Please!

One issue that needs to be addressed when using pneumatics is how to deal with the loud blast of the exhausting air created when a cylinder moves in and out.

Figure 14. Slide mechanism cylinder installed.

Figure 15. The brains and lungs.

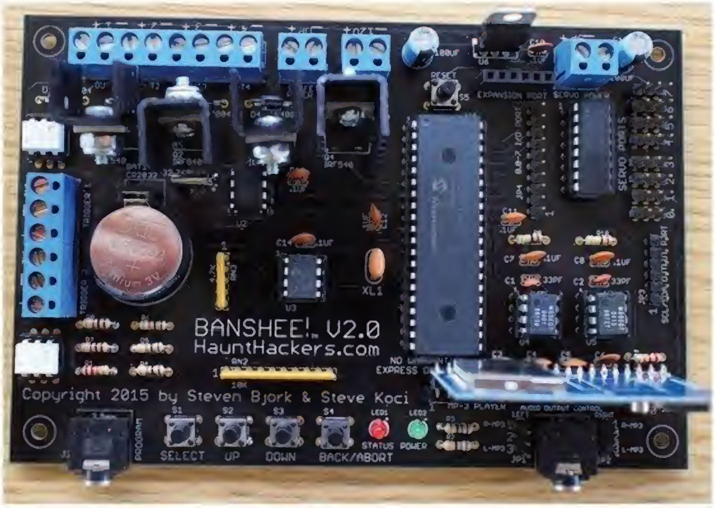


Figure 16. Banshee controller.

A Picture is Worth a Thousand Words

I would like to take this opportunity to thank my son, Bryan Koci — the owner of Red Tie Photography (**http://redtiephotography.net**) — for his marvelous work on the September 2015 project, Parkerbot. His photo was chosen for the magazine cover and he was able to bring my idea to life. It can be a time-consuming and difficult task to showcase a character in photos and video, but Bryan accomplished it in style! If you missed it, check out a back issue of *SERVO Magazine.* You can also check out his teaser video at **http://tinyurl.com/qzcyuj7**.

Figure 18. Our Pirate Pete character with pneumatic arms.

There are sound mufflers and silencers available that greatly reduce this noise, but sometimes this still doesn't silence them enough. One method I employ is to run an airline from the exhaust port to a remote location far from my guests. If this solution is not feasible, I will run the airline into the base of the prop and cover it with towels. This will help lower the sound to an acceptable level.

Choosing the proper compressor and attachments can make a substantial impact on how loud your unit is. You can further negate the noise level by placing your compressor well away from your guests or in another room or container all its own. If constructing a separate noise cancelling enclosure, be careful to allow for adequate air flow. We do not want it overheating!

Safety Above All Else

Figure 17. Slider rig made out of wood — no welding required.

I am often asked for my recommendations for plans to build pneumatic components from PVC, bicycle pumps, and washer or sprinkler solenoids. There are a plethora of plans and videos on the Internet showing how to do this, but my response is always the same: Don't do it!

It can be tempting to try to make your own and save a few dollars, but the time, effort, and cost savings are not sufficient to warrant the risk. Invest in the proper components and be assured that you are using materials that are designed for the

stresses you will be putting on them. We never want to take shortcuts that may put us or our guests in harm's way.

What Else Can I Do with Air?

How about attaching an oscillating fan to a triggered relay? This would allow you to use air to move ghosts or other characters clothed in lightweight fabrics that motion with the slightest breeze. Sometimes a simple solution is the best. We often have a tendency to make our designs overly complicated even when an easier method is available. Keep it as simple as possible. This will keep your costs down and there will be less to go wrong.

An idea I have that is just waiting for a little time to experiment with is developing a method to simulate the chest movement of a character's breathing. It would use a heavy duty balloon sealed to a rubber stopper with two holes for an inlet and outlet. Now, I just need to figure out the best way to pump in the air and then release it. I don't quite know how it will work yet, but I am anxious to give it a try.

If you've got an idea about how this might be accomplished, please share it by posting it in the forums in the new DIY Animatronics thread at **http://tinyurl.com/nvpa53g**. I would like for us to take advantage of all the incredibly knowledgeable builders we have here, so post your ideas for all to see (like the ones in **Figure 17** and **Figure 18**). I am certainly looking forward to learning from all of you!

Last Thoughts Before They Blow Away

If this sounds like something you would like to try, maybe you can add a compressor to your Christmas list and get started. You will be doing your family a huge favor by providing them with that perfect gift choice for you (that line worked with my wife as she surprised me with a fantastic new compressor, but your mileage may vary).

I am always discovering new ways to utilize pneumatics. Adding the confidence and knowledge to work with them will serve you well as you design larger and ever more complicated characters.

However, like potato chips, do not expect to be satisfied with a single pneumatic prop. You will want more! **SV**

By Jeff Cicolani

MANTIS

The Latest Offering from ServoCity

ServoCity continues to expand their robot line-up with some impressive new platforms. Their smaller robot platforms — Peewee, Sprout, and Bogie — are nice starting points for making indoor robots. Meanwhile, the Scout, Warden, and Nomad chassis are great outdoor and off-road platforms. Their latest offering — the Mantis — is another step up from their rigid and semi-rigid models. I was recently offered the opportunity to review one of these new chassis.

The Mantis is available in a four-wheel and six-wheel format. I was provided with one of the six-wheel varieties to work with and — once again — I have to say I am fairly impressed with this model. Unlike the Nomad — which is a semi-rigid chassis in that only one side pivots — the Mantis line is built using independent suspension. Each wheel and its corresponding geared motor is attached to the frame with four beams and one of their spring shocks. Being part of the Actobotics line of products, it is an impressive example of the flexibility and durability built into these parts.

The frame is constructed using two 18 inch channels attached with the open ends facing each other to create essentially a rectangular tube. This makes it very rigid. From there, each wheel is attached by four of their five inch beams. The wheels each have a 12V motor with a planetary gear box. Attachment is made using Actobotics hub adapters.

The new Mantis robot chassis from Actobotics.

These geared motors are rapidly becoming my go-to for large projects, and I've used them on several to-date. They've proven very powerful and reliable.

Aluminum spring shocks provide lift to the suspension. Each wheel assembly has one of these shocks attached. All of the moving parts were attached using aero-nuts. ServoCity (**www.servocity.com**) must have read my Nomad review in the March 15 issue, because this time all of the static parts were attached using lock washers.

Wires for the motor were cut and

Post comments on this article at **www.servomagazine.com/index.php /magazine/article/november2015_Cicolani**.

soldered onto the motor leads. The positive leads from each motor along one side of the robot were soldered together, as were the negative leads. This was repeated for the other side. They were then run through the provided grommets near the motors, through the channel, and out another grommeted hole at the top of the channel.


The center frame is two of the Actobotics 18" channels mounted together.

Attaching the needed electronics proved to be a bit of a puzzle right out of the box. The channel configuration does not provide a lot of room for electronics. Once you run the wires through the channel, it is impractical to try to get a battery and motor controller within that space. My solution was easy enough: I simply zip-tied the battery to the channel. However, this wasn't going to work for the electronics. I didn't need a lot of space since I was simply setting the Mantis up for RC control (and the receiver doesn't need a lot of space).

The RoboClaw 2x30 I ordered for the project, on the other hand, was going to need some protection. ServoCity has a mounting plate designed to attach the RoboClaw to their system, but that would leave the delicate electronics exposed. Since these off-road chassis tend to want to roll when you get as aggressive as I do with them, I've learned to keep the electronics safely ensconced in some form of protective covering. I decided to get a small project box from my neighborhood Fry's. With a little modification, I was able to fit both the RoboClaw and the RC receiver into the box. This was then zip-tied to the channel alongside the battery.

I had to cut holes in the electronics box to provide space for the large capacitors on the RoboClaw and to allow room for the servo cables. The chosen box was just

The RoboClaw 2x30 motor controller mounted in its protective box.

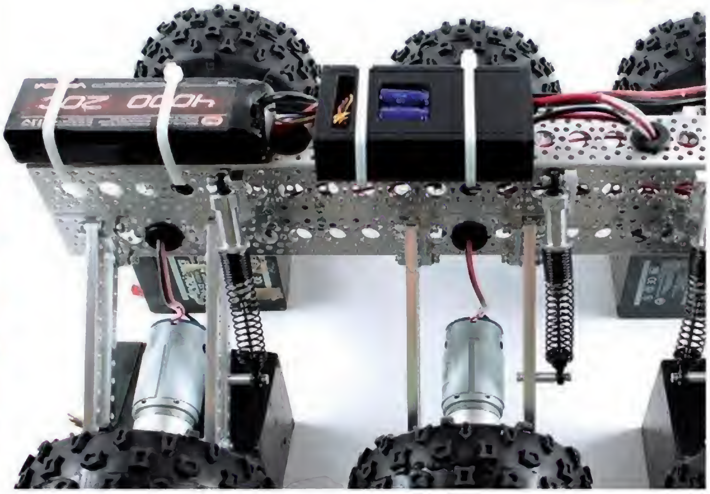

Leads from the six motors were connected into left and right channels, and then fed through a grommet on top of the chassis.

Parts Used in this Review

- Mantis six-wheeled chassis from ServoCity
- Six planetary geared 12V motors (included in the Mantis kit)
- RoboClaw 2x30 motor controller
- Venom 4000 20C 11.1V LiPo battery
- Spektrum six-channel receiver
- Spektrum five-channel transmitter

The RoboClaw motor controller with the RC receiver packed nice and tight in the protective box.

Mantis with the battery and electronics mounted with zip ties.



a little too small. This may be remedied in the future, but for now — as long as I don't roll it in a creek or let it go careening into a lake or pool — I should be fine.

The final touch was mounting the GoPro to the front. With that, it was time to take it out.

Testing the Mantis took a couple of forms. First, I ran it around the neighborhood streets to get a feel for the handling and overall performance on flat ground. As expected, the 12V motors performed excellently with the 11.1V Li-Po battery. Acceleration was good and the torque from the planetary gears had no problems tackling any of the obstacles I put it up against. In fact, the wheels lost traction long before there was any issue with the motors stalling.

Can't hit the trails without the GoPro!

There were a couple of times when the motors had a problem accelerating out of an obstacle, but once enough throttle was added it would power itself right through. However, the motors used with this kit are a known. I've used them with the Nomad and a couple other projects.

The real test was the chassis frame and suspension. In my little suburban trials, the sprung six-wheel suspension handled curbs and grass with no problem. Some adjustment was made to the front and rear springs to keep the chassis itself level while moving. When power was applied to the motors, the entire chassis wanted to lean backwards. This was not desirable when filming, so I increased the rear shocks and decreased the front. Now, it leans forward while at rest and is more or less level while traveling. Depending on future applications, I may add a pan-tilt mechanism for the camera. For these trials, however, a forward facing mount was sufficient and I wanted to keep it level during travel.

To further test the performance of the Mantis, I took it to a community park with plenty of nature trails and a couple creeks. Of course, with as little rain as we've had in the past few months, the creeks were either dry or stagnant. However, the banks were still steep, and the terrain was varied and dusty.

On the trail, the operation was smooth. Off the trail, it seemed unstoppable. Well, that is until "it" managed to get itself caught on a fallen branch. The stems got themselves hooked underneath the chassis on the motors, between the suspension arms, and the springs. This was not a big deal, but did take some work on the controls to free the bot, and some picking of stems and leaves out of joints later. After driving it through tall grass, I had to spend some time cutting grass from the wheel hubs and shaft adapters.

All of this is expected when running a robot off-road — where other robots fear to tread. None of the issues faced deterred the Mantis from doing exactly what it was told to

do. It powered through every obstacle.

One of the things I was impressed with was how nicely it crawled over irregularities and obstacles on the path. There were places on the trails with small step downs and countless exposed root systems making the path fairly traitorous for a small wheeled bot. The Mantis drove over them smoothly and effortlessly.

After the tests, I took the Mantis with me to the next Robot Group meeting here in Austin, TX. I wanted to show off this latest project and get some of their impressions. The robot caused a lot of excitement in the group. There were some questions about how it worked, how it was controlled, the effectiveness of the motors and wheels, etc. — all of which I was more than happy to answer.

There were some suggestions for using the camera and how to improve the suspension. In particular, one of our members suggested doubling the spring shocks, moving the shocks inside the struts, and replacing the screw offsets with a solid bar to provide some additional strength to these stress points. These were solid suggestions which I will look at for aesthetics if nothing else — especially since I experienced none of the issues these changes are intended to address. Save one.

There was a tremendous dust build-up at the joint where the suspension is attached to the frame. This caused that joint to seize up and the wheel would no longer move up and down. The solution is remarkably simple, however.

Since the screw that is acting as the pivot point is held on by an aero-nut, simply loosening this nut by a quarter turn should allow enough play for the dust to not collect. So, this is a fix I've implemented with good results.

Conclusion

The new Mantis chassis is a very capable platform for your next rover style robot project. The six-wheel design and independent suspension makes for a versatile vehicle on-road and even more so off-road. The only drawback I see with the chassis is no clear method for mounting electronics, but, that is what a chassis is all about: a platform on which to build. The Actobotics system provides ample mounting opportunities for adding either more Actobotics parts or your own custom supports and frame.

Like all of the robot chassis provided by ServoCity, you will want to add to the kit and likely make some modifications to suit your needs. I will be ordering some more parts from the Actobotics line to build it out a little more, and to implement some of the suggestions made by group members.

A little more play will be introduced to the bolts holding the suspension to the frame in order to reduce dust build-up and the resulting stiffness at the joints. I'll probably double up on the springs to increase the payload capabilities, and move the springs inside the suspension arms. I'll be using more channel or custom brackets to mount a platform or box on top.

In short, ServoCity has done it again with a durable, adaptable, and fun robot chassis in the Mantis. **SV**

Mantis cruising along a trail at Walnut Creek Park in Austin, TX.

Mantis tackled most terrain with little effort — especially areas like these exposed roots on the trail.

The Mantis proved indomitable on and off the trail.

Perimeter Sensing Options

By John Blankenship

Post comments on this section and find any associated files and/or downloads at **www.servomagazine.com/index.php/magazine/article/november2015_Blankenship**.

Mobile robots need perimeter sensing. There are many options, including digital and analog, IR and ultrasonic — even a laser could be in your future. Understanding the limitations and advantages of various choices can help you improve the functionality of your robot.

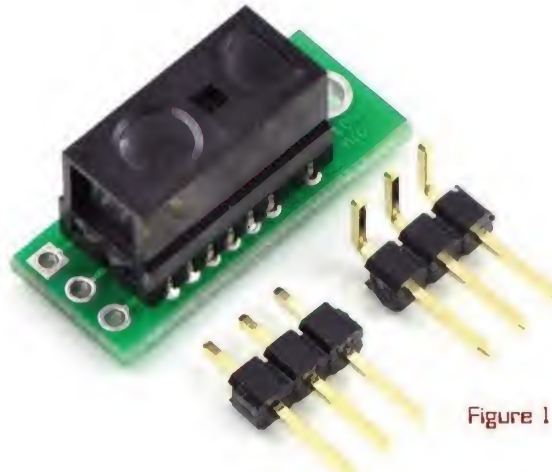


Figure 1.

Figure 2.

Mobile robots often utilize perimeter sensing in order to perform tasks such as avoiding obstacles, hugging walls, and navigating cluttered environments. Some perimeter sensors – such as the Sharp GP2Y0D810Z0F shown in **Figure 1** – are limited in that they merely provide a true/false indication of the presence of a detected object. These sensors are often referred to as digital sensors because their output is zero or one. In this case, objects can only be detected when they are within a 2 to 10 cm range.

Ranging sensors, on the other hand, are devices that can provide a distance measurement to objects that have been detected. Such sensors generally utilize infrared or ultrasonic waves to measure the distance to objects. The measured distance supplied by these sensors can take many forms. It might, for example, be a voltage whose amplitude is related to the distance measured. The data might also be provided using some form of a serial transmission (USB, I^2C, etc.). Even though serial transmissions are digital, ranging sensors are often referred to as analog sensors because the output from them is not a true/false condition.

Figure 2 shows an IR-based Sharp analog sensor (GP2Y0A21YK0F) capable of measuring distances up to 80 cm. The output from this sensor is a voltage as indicated in

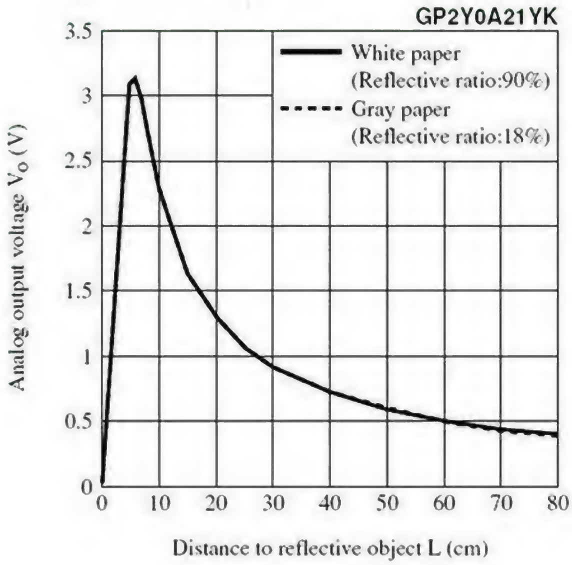


Figure 3.

Figure 4.

Figure 3, which means you must use an ADC (analog-to-digital converter) to obtain the reading. There are several things we can learn from this graph.

First, objects nearer than 5 cm can appear to be farther away because the voltage generated is the same as a more distant object. Also notice that the main part of the graph has an inverse relationship with the distance (the voltage decreases as the distance increases). Finally, observe the non-linearity of the voltage/distance relationship. This means calculations are necessary to extract the actual distance.

The Parallax PING))) sensor shown in **Figure 4** utilizes sound waves to measure distances up to three meters. In order to obtain distance readings from ultrasonic sensors, you generally must write a program that triggers the unit to send out a burst of ultrasonic sound. The program then counts the time it takes for the wave to hit a remote object and reflect back to the receiver, and then uses that time to compute the distance to the object. This calculation is linear, so it is much easier than that required for the IR sensor of **Figure 2**.

The signal indicating the wave has been received is a digital (true/false) indication, but again, these sensors are often referred to as analog because the data they ultimately provide is a variable quantity.

These two examples demonstrate why it can be more difficult to obtain ranging information than to simply detect the presence of an object with a digital sensor. When distance data is not needed, the low cost and ease-of-use of digital sensors can make them attractive — especially for less sophisticated robots or beginning hobbyists. Even ranging sensors, however, have their own limitations.

IR sensors can only sense objects that reflect light back to them, so dark objects that absorb light or objects with surfaces that greatly disperse light can be difficult to detect. IR rangers are particularly susceptible to this problem because the amplitude of the IR wave is small to begin with and it decreases with the distance being measured.

Ultrasonic sensors generally require software routines to measure the time for sound waves to bounce off objects and return to them. The linearity of the time-distance relationship makes it easy to obtain fairly accurate readings, but ultrasonic sensors can have trouble reliably detecting soft objects (such as stuffed animals, curtains, etc.) that absorb sound waves.

Both IR and ultrasonic sensors can have trouble detecting surfaces that are not somewhat perpendicular to their beams because angled surfaces can reflect the waves off to the side instead of back toward the sensor. In addition, the IR and ultrasonic sensors generally used for robotics can only reliably measure distances of 3-10 feet.

Often, an object that is difficult to detect with ultrasonics is easily seen with an IR-based sensor (and vice versa). For that reason, there are advantages for giving your robot both types so it can spot objects detected by either sensor. Parallax provides a dual-mounting bracket for IR and ultrasonic sensors (**Figure 5**) to help facilitate such an arrangement.

Figure 5.

Figure 6.

Figure 7.

Even with all their limitations, IR and/or ultrasonic ranging sensors can provide reasonably accurate (within 1/2 inch or so) and reliable information, especially in a controlled or known environment. This — plus the fact that rangers cost only slightly more than digital sensors — makes them an excellent choice for many hobbyists. Sometimes though, better accuracy, greater distances, and more reliability are needed.

Laser ranging sensors are significantly more expensive than IR or ultrasonic, but they solve many problems. Lasers generally have no trouble detecting dark or soft objects, and — perhaps even more important — they easily detect angled surfaces that can confuse IR and ultrasonic sensors. Furthermore, their detection range can be 50 times greater than IR and ultrasonic rangers.

The accuracy of a laser sensor (typically around 1 cm) is not significantly better than other types of rangers, but that's not the entire story. The readings from IR and ultrasonic sensors often bounce or jitter around the actual value. The distance to a *stationary* object might, for example, be reported as 10 inches, then 9, then 11, then 10, etc. Such individual readings are obviously not truly accurate, but an accurate reading can often be obtained by averaging the readings over time. The readings from laser rangers — by comparison — are far more stable and repeatable than other types.

For many applications, a laser's capabilities are worth the higher price. The extended range of a laser, for example, can greatly enhance a robot's navigational behaviors because it can map its environment and more precisely determine its location by accurately measuring the distance to known objects and distant walls.

Lasers don't just improve the standard mobile robotic behaviors we are all familiar with. They also make it simpler to deal with new technologies. For example, the extended range of a laser sensor can make it easy to determine the altitude of a drone, and the speed and accuracy of the measurements can enhance the ability to land autonomously.

Interfacing with many laser rangers can be difficult, but Parallax has made it easy with its new line of laser-based sensors. **Figure 6** shows a typical model in their SF10 series which can measure distances up to 100 meters. **Figure 7** shows the SF02 which has a maximum range of 50 meters. Both of these lasers were designed to be used as altimeters for small aircraft, but they also make excellent perimeter sensors for mobile robots.

No matter which model you choose, a major advantage of the Parallax lasers is their variety of interfaces which allows them to connect with almost any processor. The interfaces include USB, analog, CMOS serial, and I²C. Another advantage of these lasers is that they can be configured to meet your specific needs. Let's look at an example to illustrate this point.

I felt my Arlo robot (see the series, *The Robot*

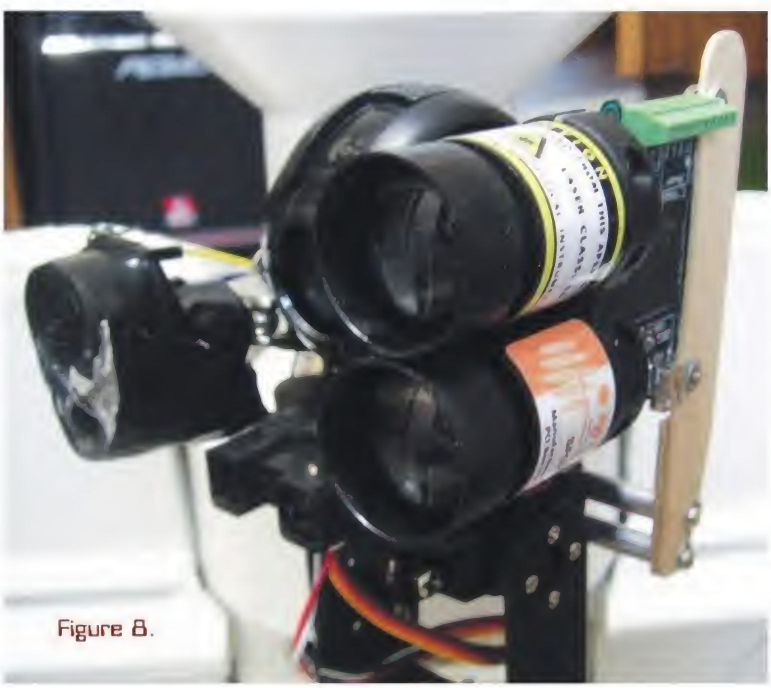


Figure 8.

You've Always Wanted, that started with the January 2015 issue of *SERVO*) could be enhanced by adding a ranging laser to his turret. For a variety of reasons, I chose the SF02. **Figure 8** shows it mounted on Arlo's turret which also supports an IR sensor, an ultrasonic sensor, a beacon detector, and a webcam. My goal was to replace the IR sensor with the SF02.

Arlo's original IR sensor produced an analog output, so I was pleased that the SF02 also had an analog option. Arlo's original software obtained the distance data with a 10-bit ADC, but the number was scaled to eight bits. I was worried that eight bits would not be enough for the laser's greater range and accuracy, but the SF02's ability to be customized made it easy to solve the problem.

Parallax provides a program called the Lightware Terminal that lets you configure the SF02 in a variety of ways as detailed by **Figure 9**. The configuration options available on the SF02 proved to be exactly what I needed.

The SF02 outputs a voltage between 0 and 3.3 volts (with 10 bits of precision) to indicate the distance measured. You can configure this output so that a full reading of 3.3 volts indicates *any* valid distance. If the analog output is read with a 10-bit ADC, the resolution can be exceptional, but obviously it won't fit in the eight bits needed to maintain compatibility with my original application.

I determined that a resolution of one inch was acceptable for my needs, which means that an eight-bit byte could hold distance measurements up to 255 inches. This distance (more than 21 feet) was also deemed acceptable for an indoor robot such as Arlo. Therefore, I set the laser's maximum distance to 6.22 meters (see **Figure 9**) which is 255 inches.

Most ADCs assume their input voltage has a range of 0V-5V. Remember, though, that the output from the laser is only 0V-3.3V. Since 3.3/5 is essentially 2/3, we can effectively make the voltage a full range reading if we multiply it by 3/2. The reading will still be 10 bits, though, so we can shift it right twice (or divide by 4) to create the desired eight-bit number.

Both of these operations together simply multiple the acquired reading by 3 and divide that total by 8. This action proved to produce an extremely stable reading with reasonable distance and accuracy. It also shows the advantage of being able to configure the maximum range of the laser.

If your robot needs more accuracy or to be able to measure greater distances, there are easy modifications. Making the maximum range 12.44 meters will let the laser measure more than 40 feet, but each unit in the eight-bit answer represents two inches.

Reducing the range to 3.11 meters will improve the accuracy to 1/2 inch, but you will only be able to measure out about 10 feet. All things considered, the 6.22 meter range and one inch accuracy seems to be a good tradeoff.

```
Zero datum offset........ 0.00 m
Smooth output............ OFF
Analog 3.3 V distance.... 6.22 m
Analog 0.0 V distance.... 0.00 m
Alarm 1 distance......... 10.00 m
Alarm 2 distance......... 10.00 m
Alarm 3 distance......... 10.00 m
Alarm hysteresis......... 0.05 m
Serial port baud rate.... 19200
I2C bus address.......... 0x55
```

Figure 9.

Of course, if you want to use a two-byte value for the laser reading, you can get both longer distances *and* improved accuracy.

The ability to configure Parallax's laser sensors should not be taken lightly. As **Figure 9** shows, you have many options. You can specify an offset to ensure, for example, that a zero distance is reported at your robot's perimeter. You can even tell the sensor to smooth any jitter by providing you with an average of multiple readings. My experience with the SF02 showed no need for the averaging, but it is a nice feature when extreme stability is needed.

It is worth mentioning that the light emitted from a ranging laser can damage your eyesight if you look directly into the beam. For that reason, you should carefully evaluate the safety needs of any project that utilizes a laser.

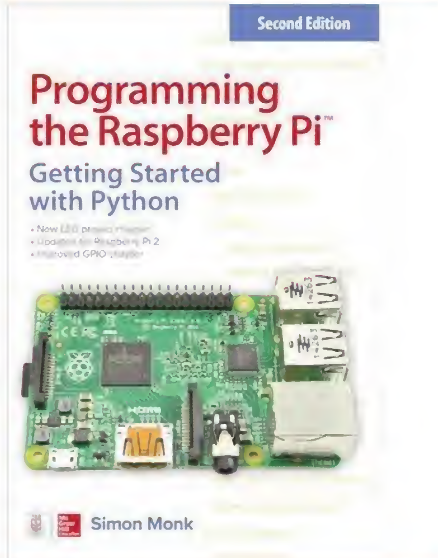
The higher cost of adding a laser to your robot does not make sense for every application, but it is a viable option when your situation requires longer distances and improved reliability.

Knowing the advantages and limitations of various perimeter sensing options can help you improve the functionality of your next robot. **SV**

The SERVO Webstore

CD-ROM SPECIALS

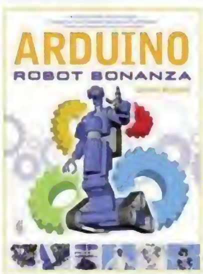
11.2015 FEATURED



ROBOTICS

Arduino Robot Bonanza

by Gordon McComb



Create high-tech walking, talking, and thinking robots. "McComb hasn't missed a beat. It's an absolute winner!"

Breathe life into the robots of your dreams — without advanced electronics or programming skills. *Arduino Robot Bonanza* shows you how to build autonomous robots using ordinary tools and common parts. Learn how to wire things up, program your robot's brain, and add your own unique flair.

$29.95

Build Your Own Quadcopter: Power Up Your Designs with the Parallax Elev-8

by Donald Norris

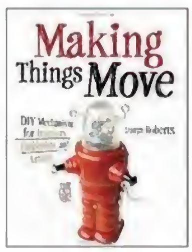
Build a custom multirotor aircraft!

Build and customize radio-controlled quadcopters that take off, land, hover, and soar. *Build Your Own Quadcopter: Power Up Your Designs* with the Parallax Elev-8 features step-by-step assembly plans and experiments that will have you launching fully functioning quadcopters in no time.

$50.00

Making Things Move: DIY Mechanisms for Inventors, Hobbyists, and Artists

by Dustyn Roberts



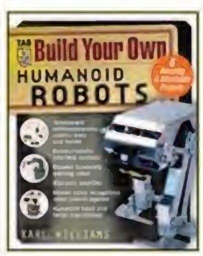
In *Making Things Move: DIY Mechanisms for Inventors, Hobbyists, and Artists*, you'll learn how to successfully build moving mechanisms through non-technical explanations, examples, and do-it-yourself projects — from kinetic art installations to creative toys to energy-harvesting devices. Photographs, illustrations, screenshots, and images of 3D models are included for each project.

$29.95

Build Your Own Humanoid Robots

by Karl Williams

GREAT 'DROIDS, INDEED!



This unique guide to sophisticated robotics projects brings humanoid robot construction home to the hobbyist. Written by a well-known figure in the robotics community, *Build Your Own Humanoid Robots* provides step-by-step directions for six exciting projects, each costing less than $300. Together, they form the essential ingredients for making your own humanoid robot.

$24.95

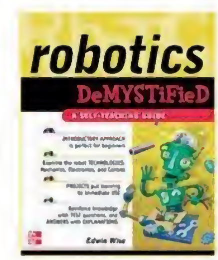
Build Your Own Teams of Robots with LEGO® MINDSTORMS® NXT and Bluetooth®

Plan, design, assemble, and program robot squads that communicate and cooperate with each other to accomplish together what they can't do individually. Inside the authors show you how to construct a team capability matrix (TCM) and use the Bluetooth Robotic-Oriented Network (BRON) so your robot teams can share sensors, actuators, end effectors, motor power, and programs. **$30.00**

Robotics Demystified

by Edwin Wise

YOU DON'T NEED ARTIFICIAL INTELLIGENCE TO LEARN ROBOTICS!



Now anyone with an interest in robotics can gain a deeper understanding — without formal training, unlimited time, or a genius IQ. In *Robotics Demystified*, expert robot builder and author Edwin Wise provides an effective and totally painless way to learn about the technologies used to build robots! **$19.95**

We accept VISA, MC, AMEX, and DISCOVER
Prices do not include shipping and may be subject to change.

Order in our online store at www.servomagazine.com or call 1-800-783-4674

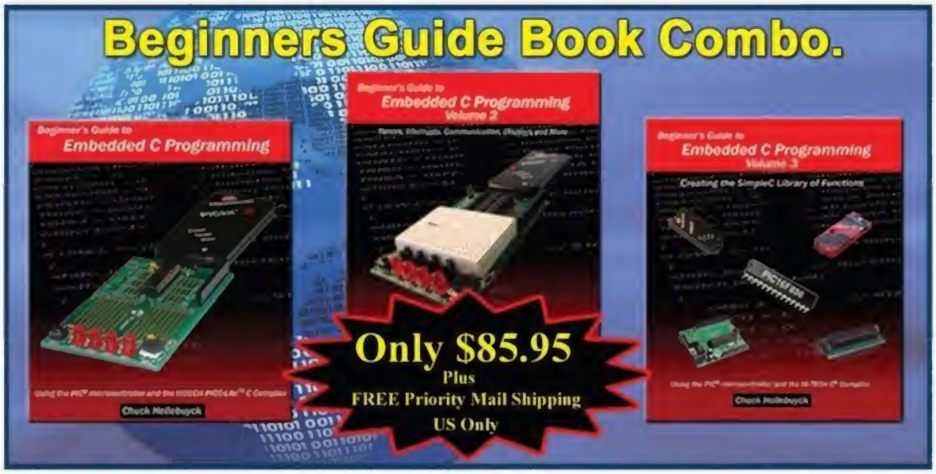
SPECIAL OFFERS!

Forbidden LEGO

by Ulrik Pilegaard / Mike Dooley

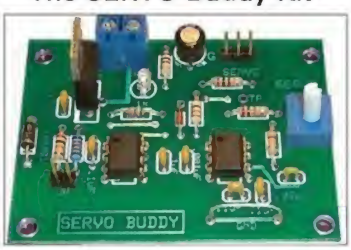
Forbidden LEGO introduces you to the type of free-style building that LEGO's master builders do for fun in the back room. Using LEGO bricks in combination with common household materials (from rubber bands and glue to plastic spoons and ping-pong balls) along with some very unorthodox building techniques, you'll learn to create working models that LEGO would never endorse.

Reg $24.95 Sale Price $19.95



PROJECTS

The SERVO Buddy Kit



An inexpensive circuit you can build to control a servo without a microcontroller.

For more information, please check out the May 2008 issue or go to the SERVO webstore.

Includes an article reprint.

$39.55

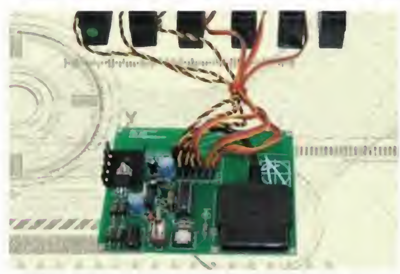
3D LED Cube Kit

From the article "Build the 3D LED Matrix Cube" as seen in the August 2011 issue of *Nuts & Volts Magazine*.

This kit shows you how to build a really cool 3D cube with a 4 x 4 x 4 monochromatic LED matrix which has a total of 64 LEDs. The preprogrammed microcontroller that includes 29 patterns that will automatically play with a runtime of approximately 6-1/2 minutes.
Colors available: Green, Red, Yellow & Blue.
Jig and plastic cases also available.

$57.95

PS2 Servomotor Controller Kit

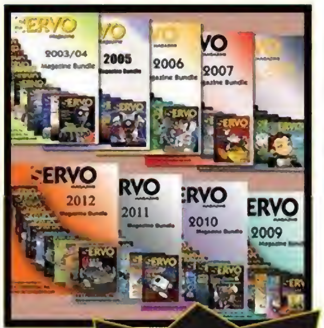


This kit accompanied with your own PlayStation controller will allow you to control up to six servomotors. Includes all components and instruction manual.
For more information, please see the February 2011 edition of *SERVO Magazine*.
Assembled units available!

$79.95

To order call 1-800-783-4624
Or Online At www.servomagazine.com

SERVO Magazine Bundles



Save $10.00 Only $57.95!

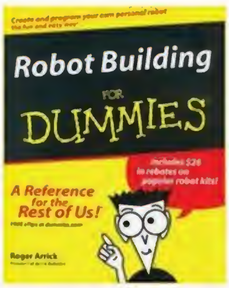
Now you can get one year's worth of all your favorite articles from *SERVO Magazine* in a convenient bundle of print copies. Available for years 04, 05, 06, 07, 08, 09, 10, 11 and 2012.

Any bot builders out there? Get cool robotics stuff for the holidays!

Call me at my order desk!

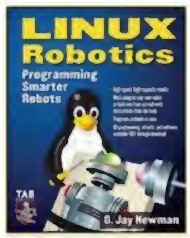
Visit my online store @ www.servomagazine.com

Robot Building for Dummies
by Roger Arrick / Nancy Stevenson



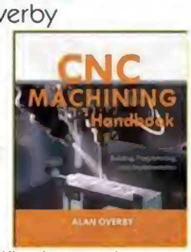
Discover what robots can do and how they work. Find out how to build your own robot and program it to perform tasks. Ready to enter the robot world? This book is your passport! It walks you through building your very own little metal assistant from a kit, dressing it up, giving it a brain, programming it to do things, even making it talk. Along the way, you'll gather some tidbits about robot history, enthusiasts groups, and more. **$24.95**

Linux Robotics
by D. Jay Newman



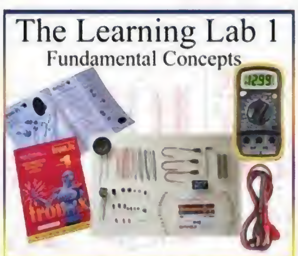
If you want your robot to have more brains than microcontrollers can deliver — if you want a truly intelligent, high-capability robot — everything you need is right here. *Linux Robotics* gives you step-by-step directions for "Zeppo," a super-smart, single-board-powered robot that can be built by any hobbyist. You also get complete instructions for incorporating Linux single boards into your own unique robotic designs. No programming experience is required. This book includes access to all the downloadable programs you need. **$38.95**

CNC Machining Handbook: Building, Programming, and Implementation
by Alan Overby



The *CNC Machining Handbook* describes the steps involved in building a CNC machine and successfully implementing it in a real world application. Helpful photos and illustrations are featured throughout. Whether you're a student, hobbyist, or business owner looking to move from a manual manufacturing process to the accuracy and repeatability of what CNC has to offer, you'll benefit from the in-depth information in this comprehensive resource. **$34.95**

LEARN ELECTRONICS - ESSENTIALS FOR BEGINNING BOT BUILDERS



$59.95

$49.95

$39.95

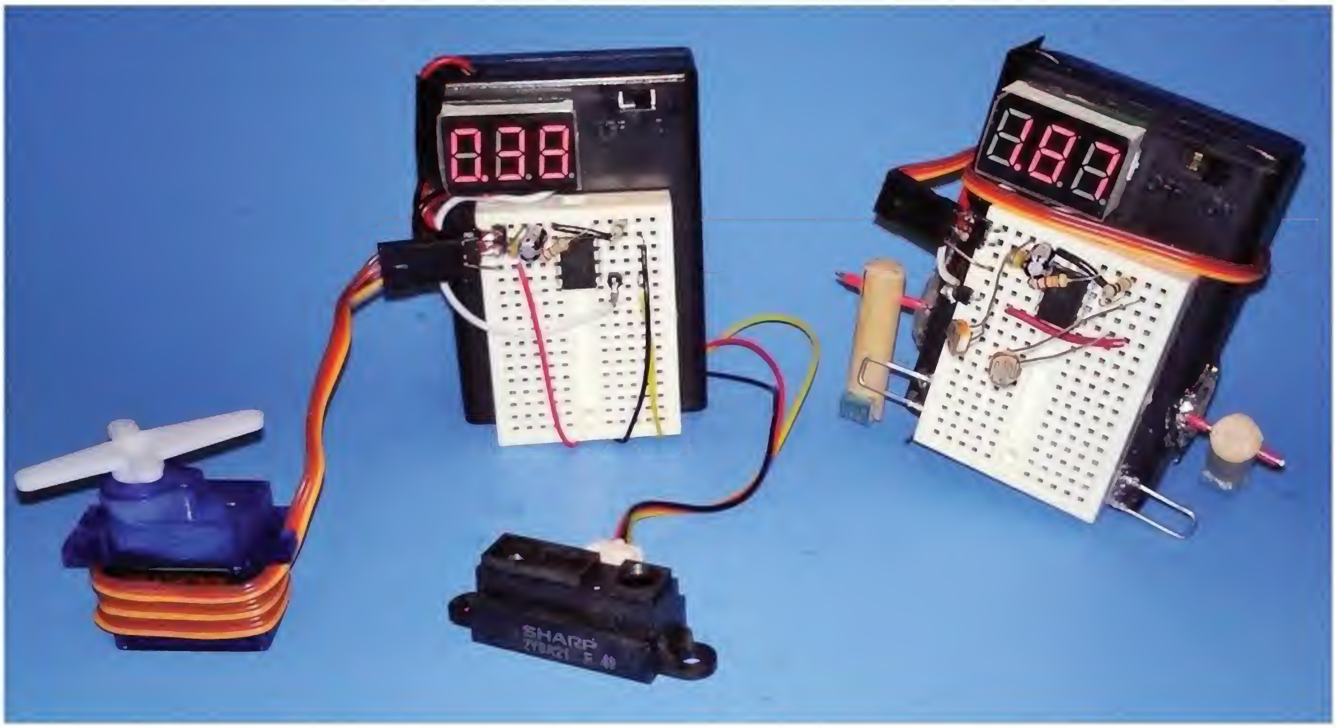
These labs from LF Components show simple and interesting experiments and lessons, all done on a solderless circuit board. As you do each experiment, you learn how basic components work in a circuit, and continue to build your arsenal of knowledge with each successive experiment.

For more info please visit our webstore.

By Eric Ostendorff

BASIC Bots & PICAXE Processors

Part 3: Analog Sensors, ADCs, and a Walking Robot!



Hello world! Last month, we connected our homebuilt PICAXE 08M2+ breadboard to some simple robotics hardware and started programming. We blinked an LED and got a servo moving under IR remote control. Part 1 was about making the breadboard and USB adapter, and getting connected to a PC using the Programming Editor (PE). Programs are written in PICAXE BASIC, then downloaded to the 08M2+ (in other words, programming the 08M2+) through the USB adapter and a three-pin breadboard connection. That same USB connection lets the PICAXE send data to the PC screen using the serial terminal (F8), which is very useful for monitoring PICAXE input and output values, calibrating sensors, and general debugging of a program. This month, we'll look at some simple sensors to give our breadboard the ability to measure distance, light, and temperature. Finally, we'll make our breadboard into a simple robot!

Post comments on this section and find any associated files and/or downloads at **www.servomagazine.com/index.php/magazine/article/november2015_Ostendorff.**

The digital world of the PICAXE is all lows and highs, zeroes and ones. The deceptively easy-to-use IR receiver we used last time is actually a sophisticated little module with lots of internal circuitry optimized for receiving nicely controlled digital control signals. Of course, most things in the real world have analog values. Temperature, light, and distance are constantly changing numbers spanning a range. Fortunately for us, analog sensors for measuring these quantities are cheap and abundant (**Figure 1**).

Our PICAXE 08M2 has three built-in analog/digital converters (ADCs) on pins C.1, C.2, and C.4 (**Figure 2**) which can readily measure from 0-5 volts. Actually, they can measure up to the battery voltage; so, 0-4.5 volts in our case. **READADC C.2,B5** reads a voltage on pin C.2, scales it to a proportional eight-bit value from 0 and 255, and stores it in variable B5. If higher resolution is needed, **READADC10 C.4,W3** reads a voltage on pin C.4, scales it to a proportional 10-bit number from 0-1023, and stores it in word variable W3. (Yes Virginia, you can change the pins and variables around; those are just random examples.)

In order to measure voltages higher than four or five volts and to use variable resistance sensors, we need to understand how voltage dividers work. Google and YouTube can do a far better job than I can, so I won't waste much space here.

Two resistors in series connected across a voltage source will develop a proportional voltage at their center junction. The resistors' values determine the voltage as shown in **Figure 3**. A rotary potentiometer is a three-terminal variable resistor — much like a volume control. Connecting the pot's end terminals to our battery forms a voltage divider. The pot's movable wiper changes position as the knob is rotated — effectively splitting the total 10K resistance into two continuously changing resistor values.

As one gets bigger, the other gets smaller, and their sum always remains 10K. Thus, the voltage at the wiper can be anywhere from zero up to the battery voltage as shown in **Figure 4B**.

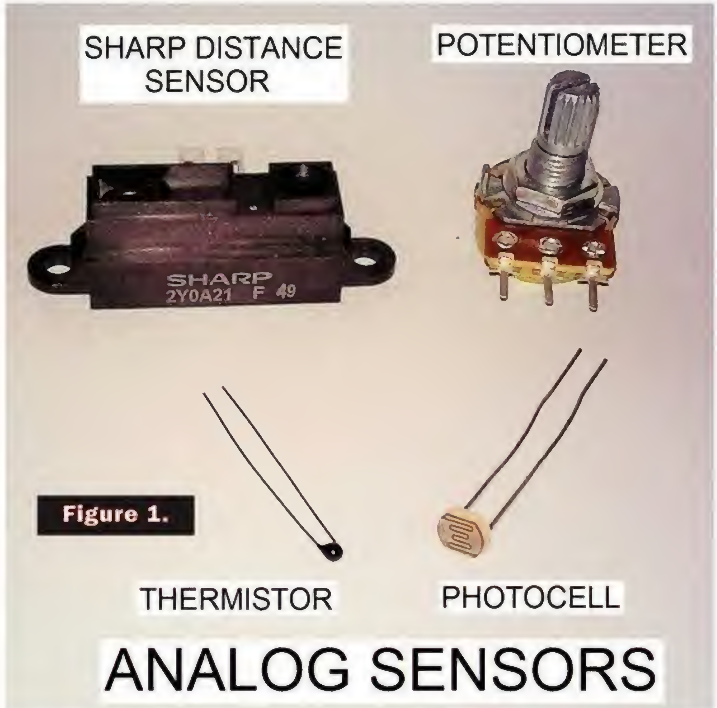


Figure 1.

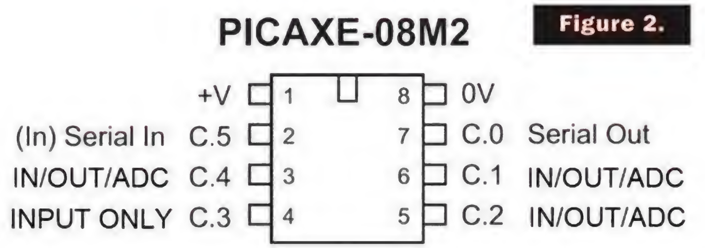


Figure 2.

C.1, C.2, C.4 ARE GENERAL PURPOSE INPUT/OUTPUT (GPIO) PINS, CAN DRIVE SERVOS AND ALSO HAVE ADC CAPABILITY. C.3 IS AN INPUT ONLY, NO ADC.

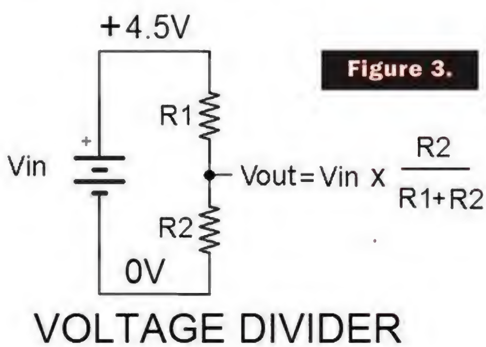


Figure 3.

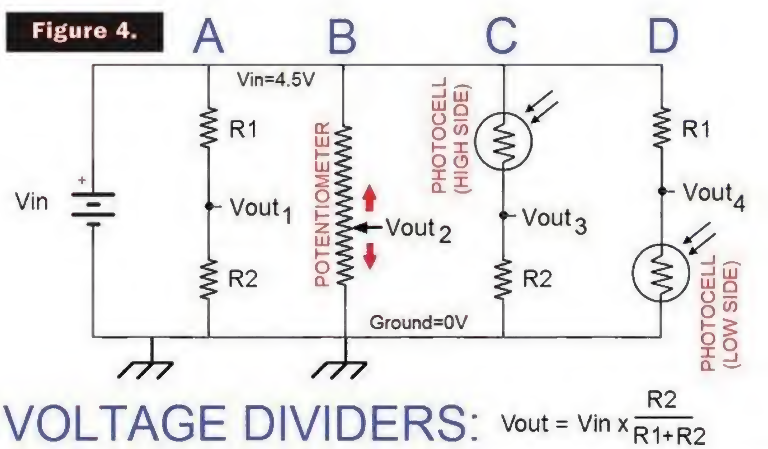


Figure 4.

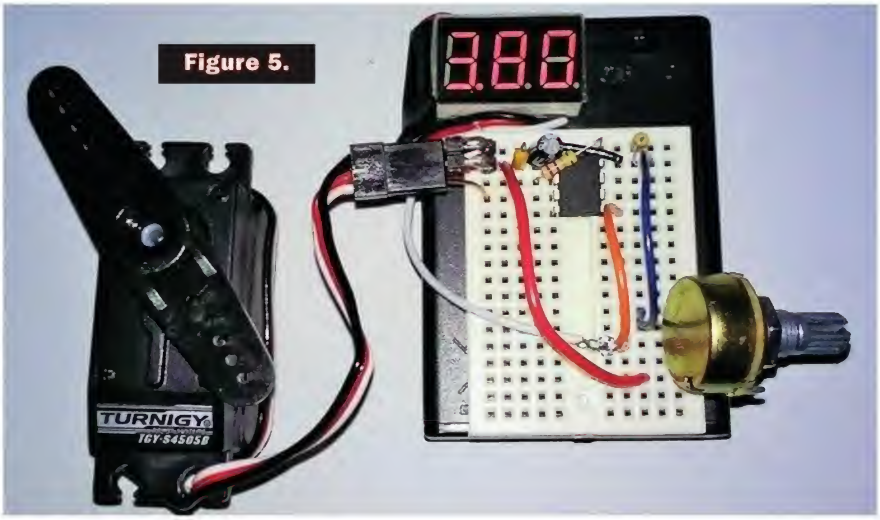


Figure 5.

```
#picaxe 08m2
#no_data
pause 100
```

A quick word about resistance values and electrical current here: 10K is a typical pot and sensor resistance value — ideal for low current applications. Five volts can only push half a milliamp through 10K ohms. That's very low current — perfect for microprocessors, but certainly not for motors or other high current devices which could damage a micro.

Use caution when connecting voltage dividers to your PICAXE — especially pots — which can be twirled down to zero ohms at each endpoint, which could "release the magic smoke" if connected improperly.

Keep in mind that there are three related but very different numbers changing together: resistance, voltage, and the **READADC** value. Besides being a manually adjustable knob, a potentiometer can be a rotary position sensor. Analog servos measure their angle of rotation with an internal potentiometer.

You might measure the angle of a robot arm joint with a potentiometer, or create a pantograph-style digitizer with a few pots on the linkage joints. Per **Figure 5**, our PICAXE makes quick work of using a pot to control a servo, as shown at **www.youtube.com/watch?v=JY-9ytlI2EE**:

```
servo c.4,150      ' initialize servo on pin c.4
do                 ' start loop
readadc c.2,b0     ' read pot value on pin c.2,
                   ' save as b0
sertxd (#b0,13,10)      ' display values on screen
servopos c.4,b0         ' move servo on pin c.4 to
position b0
loop                    ' end loop
```

Recall from previous issues that each of these program listings should be preceded by:

Photocells (a.k.a., photoresistors or Light Dependant Resistors, LDRs) are light sensors and very inexpensive: 50 pcs GL5528 for $2 — so stock up! Typical resistance varies from 1K (bright flashlight beam nearby) to over 200K in darkness. Typical room values are 2-10K.

Thermistors are resistive temperature sensors — also cheap at 5 pcs for $1! My "10K NTC 3435" thermistors measure 10K at room temperature; 28K in an ice bath; and 700 ohms in boiling water. Both photocells and thermistors will need a fixed series resistor to make a voltage divider.

A 10K fixed resistor is a good starting point, but the best fixed resistor value is generally the same as the value of the sensor at the center of the anticipated range.

Note that a fixed series resistor will limit the voltage range and reduce the range **READADC** values. It's helpful to note that photocells, thermistors, pots, and resistors have no polarity, and can be hooked up in either direction.

Figures 4C and **4D** show two opposite ways to make a voltage divider using a photocell (or thermistor). Method C will increase voltage and ADC values with increasing light (or temperature); method D will decrease values with increasing light (or temperature).

Figure 6A.

Photocells are fairly directional as they catch the most light and have the lowest resistance when they point directly at the light source. We can make a directional light sensor by using two photocells in series, aimed 80 degrees apart as shown in **Figures 6A** and **6B**. This differential light sensor does an analog comparison of two different light levels.

When both photocells face a light source equally, their center voltage is half of the battery voltage, but the voltage will move toward zero or full battery voltage as the light moves to favor one of the

photocells. This is the perfect chance to use our breadboard's DVM (digital voltmeter) to see the junction voltage, so move the white wire to the photocell's center connection.

Check out my video demo at **www.youtube.com/watch?v=8gotiXIKFV8**. This uses the same ADC code given previously. Interestingly, this differential sensor can't determine the overall brightness level in the room. That requires a third photocell joined to a fixed resistor (as previously described) and another ADC input.

Distance is not reliably measured by a simple resistance sensor. Sharp makes sophisticated PSD (position sensitive device) distance sensors which use reflected IR, are amazingly accurate, and output an analog voltage from 0.4-3.2V.

Our PICAXE ADCs can easily measure these voltages. Sharp's model GP2Y0A21 is a typical version, measuring 10-80 cm and priced from $7-$10 (with cable). Per **Figure 7**, its voltage output is non-linear and inversely related to distance.

Polarity does matter with this sensor, and the three-wire cable is color-coded. The yellow output wire goes to the ADC pin as shown in the opening photo.

A My Sharp sensor demo controlling a servo is at **www.youtube.com/watch?v=zsFE5lIQJs4**. The sensor's analog output fluctuates a bit using 4.5V of alkaline batteries, making the servo a bit glitchy.

Sharp recommends adding a 10+ µF filter cap at the sensor, but instead I did a software filter and averaged 20 sensor readings before moving the servo, which helped considerably:

```
servo 4,150       ' initialize servo on pin 4
do                ' start loop
w1=0              ' zero average total
for b1=1 to 20    ' loop 20 times
readadc 2,b0      ' take ADC reading
w1=w1+b0          ' add all readings together
next              ' end loop
b0=w1/20          ' divide total by 20
servopos 4,b0     ' update servo position
loop              ' end loop
```

Now, we have the tools to make a simple robot. Using just one servo, we can make an ultra-simple mechanism to make our breadboard walk. It uses short passive legs which nudge the robot forward as the servo alternates lifting the left and right sides of the robot. It's clunky but cute. See the video at **www.youtube.com/watch?v=AR-4iANt37E**.

You can see that I'm a scratch builder who favors commonly available parts and copious amounts of Super Glue™ (thick, gap filling Super Glue and Zip Kicker are my idea of "rapid prototyping").

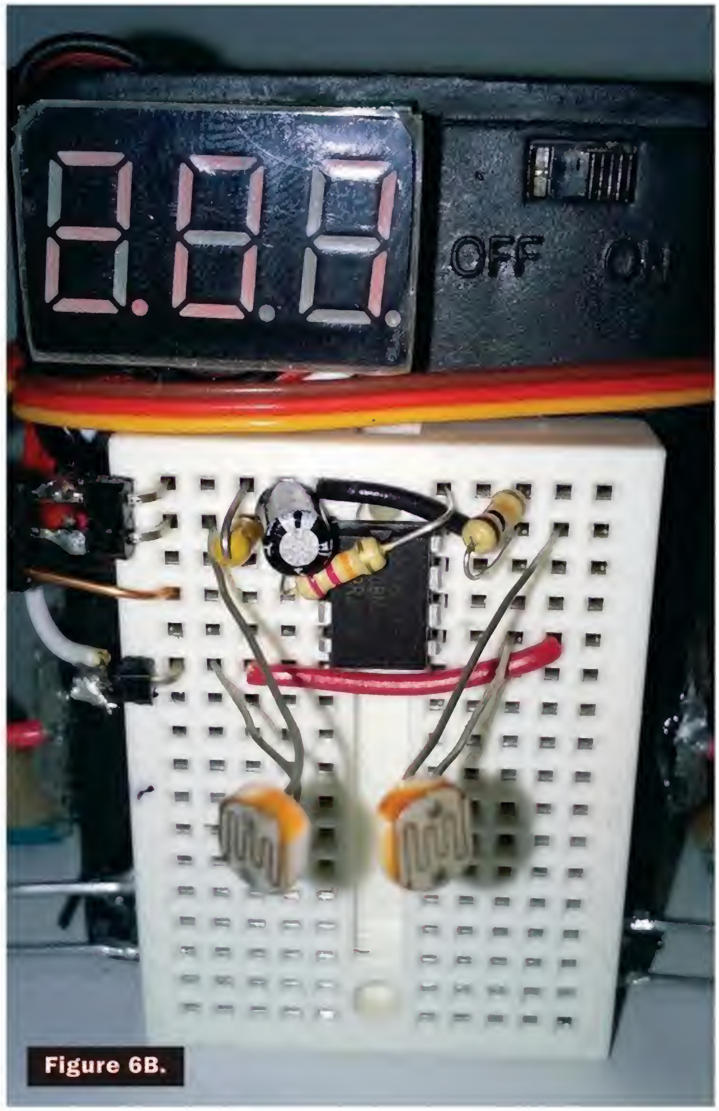
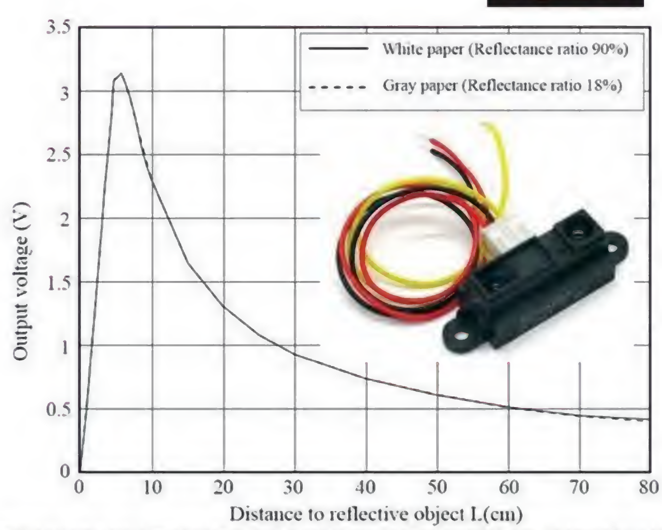


Figure 6B.

Figure 7.



SHARP GP2Y0A21 SENSOR OUTPUT VS DISTANCE

Figure 8.

Figure 8 shows the parts I used, but robotics is about improvising, so feel free to use what you have. A large paper clip is a perfect fit in the servo horn holes. Make sure the servo is in its center position when the horn and paper clip are horizontal. Per **Figure 9** and **Figure 10**, the servo is glued to the battery door just above ground level with a scrap plastic spacer. Bend the paperclip around the battery box to make forward-facing horizontal feet on each side. The feet should both touch the ground at the same time, and just clear the battery box when swinging. An additional video with details is at **www.youtube.com/watch?v=_99Bq5mOFDs**.

I used T-pins for my leg axles since they are stiffer than paper clips; music wire would work too. The axles are parallel to the ground; about 0.900" above ground level. The legs are 5/16" dowel rods, drilled 1/16" for the axle. The V-shaped bottom has a piece of rubber band glued on for traction. The legs are 1.4" long, 1.2" from the axle hole to the bottom of the rubber tip, so they hang down 0.3" below the bottom of the battery box. A front stop (paper clip) prevents the free-swinging legs from going vertical. This ensures the bot falls forward when the leg contacts the ground. To walk straight, the servo alternates raising the left and right sides. To turn, just one side is repeatedly lifted. The servo should only lift high enough for the legs to swing forward. A simple program which suddenly slams the servo from side to side works, but it's jerky and uses excess battery power. Better to have software subroutines which lift each side gently and return to center.

A subroutine is a part of a program called with a **GOSUB** command. The program execution jumps to that code. A subroutine always ends with a **RETURN** command, which makes execution jump right back to where the **GOSUB** was called. Very handy!

Here's the code used in the first video where it walks straight for a few steps, turns a bit, then repeats:

```
servo 4,145       ' initialize servo center
                  ' position
do                ' start main DO loop
for b1=1 to 4     ' loop to walk 4 steps straight
gosub left        ' jump to 'left' subroutine
gosub right       ' jump to 'right' subroutine
next              ' end of this for/next loop
for b1=1 to 5     ' loop to turn 5 steps
gosub left        ' jump to left subroutine
next              ' end of this for/next loop
loop              ' end main DO loop

right:            ' subroutine for right step
for b0=145 to 170         ' loop from center to
                          ' right
servopos 4,b0     ' move servo
pause 10          ' pause 10 milliseconds
next              ' end of this for/next loop
for b0=170 to 145 step -1    ' loop from right to
                             ' center
servopos 4,b0     ' move servo
pause 10          ' pause 10 milliseconds
next              ' end of this for/next loop
return            ' end subroutine

left:             ' subroutine for right step
for b0=145 to 125 step -1    ' loop from center
                             ' to left
servopos 4,b0     ' move servo
pause 10          ' pause 10 ms
next              ' end of this for/next loop
for b0=125 to 145        ' loop from left to
                         ' center
servopos 4,b0            ' move servo
pause 10                 ' pause 10 ms
next                     ' end of this for/next
                         ' loop
return                   ' end subroutine
```

Figure 9.

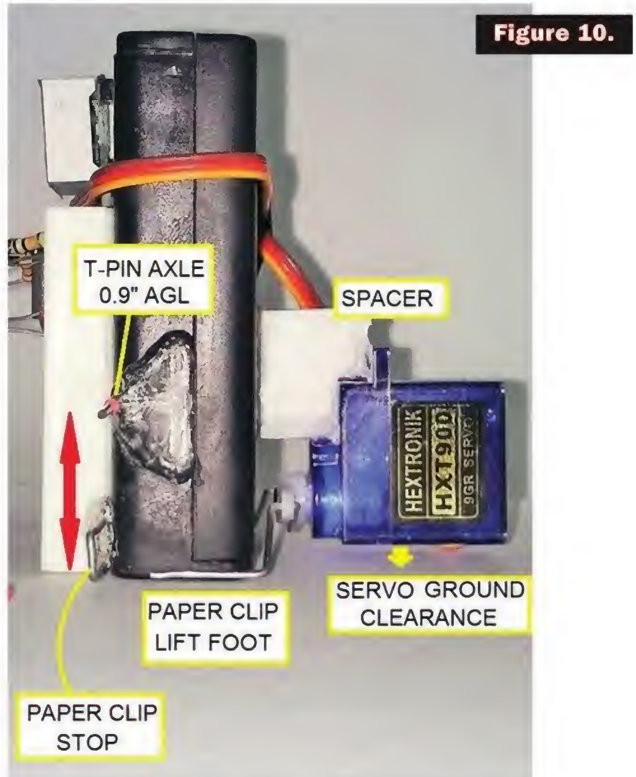


Figure 10.

Once you get that working, you can change the first part of the program to walk in any pattern you like. Of course, any good robot needs sensors, input, and a 'behavior.' It's shamefully easy to make our robot into a light seeker.

Remember the two-photocell differential light sensor? It outputs a value to the ADC indicating where the brighter light is. Above 127 means left and below 127 means right. Since our walker rotates as it walks, the values are always changing — even when it's generally headed straight for a bright light.

So, our whole program is: If the light is on the left, step right, and vice versa. It works surprisingly well (check it out at **www.youtube.com/watch?v=4CaAq6gQbLY**):

```
do                          ' start DO loop
readadc c.2,b1              ' get light sensor value
sertxd(#b1,13,10)           ' send reading to
                            ' display
if b1<127 then gosub right   ' light left to step
                             '  right
if b1>127 then gosub left    ' light right so
                             ' step left
loop                        ' end DO loop

right:  (include right subroutine from above)
left:   (include left subroutine from above)
```

With that, I leave you to rule your robot with a flashlight. Play around and see what else you can do. You could make it IR remote controlled with the info from Part 2. You could add the Sharp distance sensor on top and make a gesture-controlled robot, as I did on another robot (**www.youtube.com/watch?v=5rcjZZreLFY**).

You could add a thermistor so he walks left or right depending on the temperature ... okay, maybe not.

Whatever you decide, you are now officially a roboticist and therefore a little dangerous. What will *you do* next? **SV**

by Tom Carroll
TWCarroll@aol.com

Why are Robots So Hard to Develop?

I remember the '80s when people were excitedly talking and writing about the robot revolution that seemed to explode in that time period. In April 1984, I had the chance to attend and speak at the first International Personal Robotics Congress (IPRC) in Albuquerque, NM. A photo of some of the robots 'in attendance' is shown in Figure 1. Optimism was rampant; robots would be in everyone's home. All the biggies were there: RB Robot, Heath Hero series, Androbot, and even suppliers like Polaroid — makers of the popular electrostatic distance transducers. The thoughts were, if personal computers were becoming popular, personal robots would follow right behind them and become just as useful, and the IPRC was the proof.

Figure 1. Robots at the 1984 International Personal Robot Congress.

By 1984, there were enough personal robot manufacturers in existence and being newly formed that all of us who were interested in robotics started to believe the hype (notice I said *useful*, not popular to describe the draw to robotics). 'Popular' or 'unique' was the key in those early days, and, unfortunately, it still is. Even my management at Rockwell wanted me to dig in and find out how this new science of robotics could benefit our company since many other organizations were starting to use robotics.

Transferring this microprocessor technology to many of the robots being produced in those days seemed logical since the same ICs were used in the PCs of that time.

Going back a few years to the early '70s, the advent of microprocessors allowed far-sighted people such as Bill Gates and Steve Jobs to develop truly usable software and computers for all of us. The early 8080, 6800, 6502, and Z80 found their way into crude but useful 'computers.'

There were many steps before the even earlier ICs (such as Intel's 4004, 4040, 8008, and later versions such as the 8086, 8088, and even Motorola's 68000) became the heart of PCs. The first 'computers' such as the 8080-based MITS Altair in 1974 (shown in **Figure 2**) and the Imsai 8080 in 1975 (shown in **Figure 3**) were not particularly user friendly, as programming was accomplished by flipping switches.

Gates and his partner at Microsoft (its name when first incorporated on November 26, 1976), Paul Allen got their start in developing software for the Altair.

The Imsai gained a bit of fame in the 1983 movie, *War Games* starring Matthew Broderick and Ally Sheedy, which is shown in **Figure 4**.

Post comments on this article at **www.servomagazine.com/index.php/magazine/article/november2015_ThenNow.**

Figure 2. MITS Altair 8800 from Vintage Computer.

Figure 3. Imsai 8080.

Figure 4. Imsai 8080 shown in the 1983 movie, War Games.

Figure 5. Original Apple 1 computer PCB from 1976.

I thought it a bit amazing that this seven year old computer was featured in the film when far newer computers with actual key boards, monitors, modern processors, and useful software were so prevalent in '83. Even the 6502-based Apple 1 (shown in **Figure 5**) that was designed by Steve Wozniak and Jobs and sold by Apple in 1977 for $666.66 had a keyboard.

Should you be lucky enough these days to run across an Apple 1 in a homebrew box as in **Figure 6**, politely give the seller the $25 he's asking for it and hurry home with your prize. Find yourself a huge safe and hide it because these relics have sold for as much as $1 million.

Why was Early Robot Development Lagging Behind?

I remember one member of the Southern California Robotics Society saying in the late '70s that his 6502-based robot was proof that future robotics development had finally found the key to success. "With access to these powerful processors," I remember him quoting, "We will be able to build some amazing robots." We had just formed with the help of some NASA JPL personnel and *Robotics Age Magazine,* and were meeting at a library.

That magazine has not been published for decades now, and the group's name was changed to the Robotics Society of Southern California in the mid '80s, but the statement really struck me as a real turning point for robotics. I don't remember the guy's name and the exact words, but I was left with the feeling of being a part of the beginning.

All of our robots back in those days were strictly rolling mobile platforms, though there were some water-borne and autonomous underwater vehicles. Walking bi-pedal robots did not really exist except in labs operated by university researchers such as Marc Raibert, the founder of Boston Dynamics (now owned by Google). There were multi-legged quad and hexapods, but the balancing of a two-legged robot was beyond most experimenters as gyros and accelerometers were very expensive.

Figure 6. Apple 1 computer with a fancy case.

What Technology was Available in 'the Beginning?'

One of most prevalent mechanical parts that is used by so many robot experimenters these days is the model aircraft servo. We had them 'back then,' though they were analog types and the varieties were limited. I saw a few builders who used model aircraft servos for small robots — both modified for continuous rotation and as-is — but the small sizes available were not conducive for larger robots. Most of the motors used for robots were brushed type gearmotors that were found at surplus places, or electric window motors from automobile parts stores.

Surplus motors of all types were available at many war surplus places and from catalogs. Leftovers from the Korean and Vietnam wars were a goldmine for robot builders. Do you remember when there was no Internet and you mailed in an order form from a catalog? I was fortunate to live 25 miles from C&H Sales up in Pasadena, CA and probably bought over $1,000 in motors, gears, actuators, and all sorts of 'robot stuff.'

Up until that time, C&H had a great quarterly catalog and an even greater selection at the store itself. The catalog had detailed specs on the motors and the store personnel were just as knowledgeable. These days, sometimes the best value or just the cheapest items are made in China as surplus items are not as prevalant.

What Factors Make Robots Difficult to Develop?

One major factor is the general public's impression of the state-of-the-art in the science and applications of robots. *The Six Million Dollar Man* TV series from the 1974 time period featured test pilot, Steve Austin who was severely injured after a crash of a test spacecraft. The opening scene of each episode showed an actual crash in 1967 of a manned M2-F2 lifting body in which the pilot was injured and lost vision in his eye due to an infection. For some reason or another in the TV series, the Air Force decided to resurrect Austin with a new set of legs, a bionic arm, and a bionic eye. "We can rebuild him. We have the technology. We can make him better than he was. Better, stronger, faster."

Of course, he had nuclear power generators, and the series was so successful that the producers brought forth a bionic woman. The ability to run over 60 MPH, lift a car, and leap from the ground to the top of a building was due to the internal nuclear power. All this was portrayed as actual science over 40 years ago. We're not even close to this technology today.

"If it is on TV, it must be true," viewers would say. "The producers would not stretch the truth and depict something that is not available to the military." Today's *Extant* TV series depicts a 'humanich' boy who looks identical to and acts just like any eight year old would.

The CGI-created NS-5 robots in the movie, *iRobot* were to be in everyone's home to perform all sorts of tasks. C-3PO, R2D2, Chappie, and every movie and TV show with 'robots' depicts this technology to be far more advanced than anything in actual existence. Despite the fact that virtually every adult who sees these movies knows that they are a fantasy created by the producers, deep inside they immerse themselves into the film and visualize the robots as real.

Developers Try to Create Products to Satisfy Buyer's Desires

It is hard to sell a product to a potential customer when they have an opinion of what the product should be. I have heard people express opinions at robotics expositions and contests such as at the Robothon in Seattle Center sponsored by the Seattle Robotics Society. "What can the robot do?" many ask. After hearing that question and supplying an answer, the builder can imagine the thoughts of the person asking that question. The builder can see the reaction in the person's face: "Oh, is that all it can do? I've seen robots in movies and on TV that can do a lot more things."

You can spend hours talking to the person about all the advances made in robotics in the past decades, but you'll never completely change their views. Public opinions are difficult to change without an amazing demonstration of something new.

A Robot Builder Describes Design and Programming Challenges

I read a couple of interesting articles from Robotics Trends this past July and August entitled, *4 Challenges Holding Back Robotics* and *4 Reasons Programming Robots is Difficult* — both by Jason Ernst of Redtree Robotics. In the articles, he states: "When people see movies about the future, they see robots that interact with their environment, learn quickly, and adapt to changing conditions."

Though he is comparing fictional to existing robotic products, potential buyers of personal robots want this cooperative interaction. He mentioned fragmented platforms that each manufacturer developed for its own product — none of which were compatible with another.

Ernst also described the difficulty of programming as another stumbling block for a user. This, of course, reflects back on the robot's designer. "Robots are not plug-and-play like computers were in the beginning," he stated. "There is also a lack of built-in reliable communications. In robotics, most people use Wi-Fi or something similar." Ernst mentioned one of the biggest factors in available programming and software is the expense in developing affordable

platforms to utilize this software.

This difficulty reminds me of PC's early days with form factors such as the S-100 bus used by the Intel 8080-based Altair. I still have a stack of blank S-100 boards from the old days that I bought 'back then' thinking they would be the industry standard expansion bus. Nope. That was not to be. Ernst also described interoperability and how wireless connectivity is always an industry afterthought. "None of the products you see today or in the near future are built to work with one another," he stated.

Stumbling Blocks in Mobile Robot Design

Every robot designer has his or her own trouble area of robot design. Certain limitations can restrict a final design to something less than a designer would desire. Mobile robot design presents the builder with a whole new pile of potential problems — from reliable batteries to energy-efficient (but powerful!) motors.

Of course, the design process depends on the expected use of the robot, the complexity level of the intended final design, and funds and time available for the parts and construction.

Better Robot Power Sources are Needed

Let's discuss a few of the system and component improvements, in my opinion, that are needed for tomorrow's robots. In my experience, the power source is the main limitation for most mobile robot designs. In the beginning, we had lead-acid batteries for large robots and 'flashlight' batteries for smaller creations. Lead-acid batteries could leak sulfuric acid and were very heavy — even the gelled-electrolyte SLA cells. NiCds were the next step for size and power density, and became popular for mid-sized robots. They still presented a few problems, however, such as limited capacity and a charge/discharge 'memory effect.'

NiMH batteries came along next with greater power density and no memory effect, but even more power density was still required.

The newer lithium series of batteries seemed to be the answer, but they have the nasty habit of catching on fire — especially lithium polymer batteries. A designer may look at a battery and see a high number of amp-hour capacity, and fail to realize that most batteries — particularly lead acid — are specified in a 20 hour discharge time. When their design sucks all the power in an hour, the batteries will supply power at a much lower amp-hour rate. Think about it.

So, what do robot developers have available for their creations? Many robots draw a tremendous amount of power and can suck a battery down in less than an hour. Honda's great little Asimo can barely last an hour on its huge backpack battery. Even the $1 million+ Atlas robots used in the DARPA Robotics Challenge (DRC) needed huge batteries that lasted less than an hour.

Yes, bipedal humanoids draw more current than most mobile robots do, mainly because of the many servos/motors in the various joints, but all mobile robots are power hungry. An indoor robot traversing smooth floors does not require as much power as an outdoor robot traversing grass or uneven ground, but the bottom line is: All robots gobble electricity like a participant in a pie-eating contest. We need dramatically better battery technology for today's robots.

Contact with the Outside World

Our electro-mechanical friends can see their environment in a similar way as we humans do through sensors. We have eyes, ears, tactile senses, semi-circular canals in our ears that act like gyros, positional feedback in our extremities, plus senses of smell and taste. Despite what is seen in the movies, our organic sensors are far superior, having been honed over millions of years.

In the *Terminator* movie, we see what is supposedly being viewed by the human-robot, as rows of words in its vision. The robot can delve into its extensive database to determine that a particular 18-wheeler that it's looking at has a 10 speed transmission and a 250 HP engine, then downloads a trucker's driving course in five seconds. Our human database is our educated brain.

Our eyes are quite superior to any man-made vision sensors, and connected with our brain make an amazing sensor system. The same thing applies to our ears. We do not require any special hardware and/or software to compare visual images with a series of patterns in a separate processor. The 'processor' just happens to be a different part of our brain.

In reality, even our most advanced offerings such as the five Atlas robots used in the recent DRC tripped getting out of an ATV, tripped while walking on rubble — all while being controlled by intelligent humans. They all possessed advanced vision systems, but they can only determine so much about their environment. Don't get me wrong. I am amazed at what all of the entrants at the DRC accomplished, but this competition really opened the eyes of robotics experts, as well as the general public. Today's robots do have many limitations.

Robot Sensors Become Affordable

It may seem I have used disparaging words to describe today's MEMS gyros, compasses and accelerometers, vision cameras, and other modern sensor systems, but that was not my intent. In past times, we might have stood on a golf course and estimated that the 12th hole pin is 'about' 180 yards away, whereas a robot with today's sensors standing beside us would know it was actually 201.2 yards away. Yes, there are many types of handheld range finders designed for golfers and hunters, but these are read by *humans* viewing

Figure 7. Pololu AltIMU-10 v4 gyro, accelerometer, compass, and altimeter.

images and digital range figures on a screen in the instruments.

Hackers used microcontrollers in the past to read the digital information from these handheld range finders for robot distance measurements before 'made for robots' sensors became widely available. Today's very inexpensive micro electro-mechanical systems (MEMS) have allowed robots, quadcopters, and many moving platforms to sense orientation and motions, and be easily controlled. The latest MEMS technology has allowed complex navigation and appendage motions to be under total control of a microprocessor. Shaft encoders, accelerometers, gyros, and magnetic compasses have plummeted in cost from many hundreds of dollars to just a few bucks. The Pololu AltIMU-10 shown in **Figure 7** combines a gyro, accelerometer, and a compass IMU with an altimeter on a single tiny board. These MEMS chips have allowed robots to possess some amazing mobility and mobile apabilities.

We have also made amazing strides in other robot sensor technology in the last few years. Budget-minded robot builders have access to long-range laser range finders such as the LightWare SF-10 series shown in **Figure 8** from Parallax that costs between $350-$550 and can accurately measure distances from 25 meters up to 100 meters. This amazing 'time of flight' sensor can replace far more expensive rotating laser systems for many robot applications.

There are also now 'intelligent' video cameras designed for robots such as the CMUcam series costing from $139 to $259; the CMUcam3 module is shown in **Figure 9**. These cameras have proven very functional and quite useful for mobile robots.

What are the Main Stumbling Blocks in Robot Design?

I have been writing about all the amazing (yet inexpensive) sensors available to today's robot designers. We can talk to our robots and they can talk back, as well as see and follow us and obey our commands. Batteries can become a major problem when building human-sized bipedal robots, but not as much for smaller rolling mobile platforms. Software such as ROS, Python, C++, Java, Microsoft RDS, PBasic, Linux, and many other systems are available for builders using a Propeller, Arduino, Raspberry Pi, Beagle Bone, or any other type of microcontroller. So, what's the problem here?

Let's look back at those early computers again. They exploded in popularity when the CPU board was placed in the same 'box' as a floppy drive, and affordable and useful software was available on a floppy disc. The computer's power supply was also an integral part, and the monitor could sit on top or be contained in the same case. Later versions added a hard drive, and we now had a computer that did everything we could imagine. We could play games, develop spreadsheets, and perform word processing with the addition of a cheap printer instead of using typewriters (on which the words could not be changed without an eraser or white-out). Home users and businesses gobbled them up by the millions as processor power increased, as well as the capabilities of various types of software.

Think about the earliest robots. The first personal and experimental robots started out small and grew larger. Boxes with little more than wheels were considered robots. Then, model aircraft servos were modified for continuous rotation for small one pound robots such as the Parallax Activity Bot and other similar experimental and educational robots. The larger commercial home robots shown back in **Figure 1** that were destined to change the world (but did not live up to their hoped-for popularity) like the Hero 1 and 2000, RB5X, Topo, and the others, were cute but could really do little else. The Hero series and RB5X had arms, but they were quite inefficient and sufficient only for experimentation.

Figure 8. Parallax LightWare F-10A distance sensor.

Figure 9. CMUCam3 intelligent camera for robotics.

Mechanical Aspects of Robot Design

As I have discussed so many times in this column, the mechanical design of a robot seems to be the designer's greatest stumbling block. It seems to be fairly easy for a beginner to take two modified servos and mount them to a small plastic or metal platform; add two wheels to the servos and a caster for a third wheel; add a Propeller, BASIC Stamp, or Arduino microcontroller board and some AA cells ... and you have a robot. This is a great way to start learning about robots.

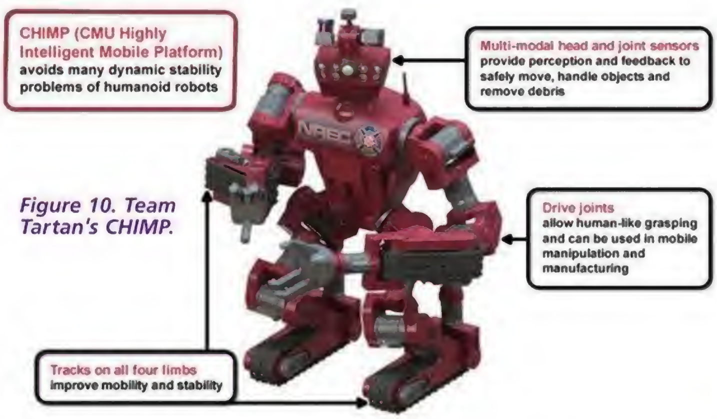


Figure 10. Team Tartan's CHIMP.

Not long after, a person wants to go bigger with a robot that can do more than just move around and respond to its sensors. The builder wants a robot that can "do something" when he/she is asked about the bot. Now all of a sudden, servos have to be replaced by gearmotors, and larger batteries are required. Mounting sensors and vision systems require special bracketry. Larger motors also need very secure mounting techniques for the drive systems. The addition of an arm or two, and the builder is faced with new metal structural needs. Cutting and bending sheet metal or machining metal becomes a bit scary for many folks.

A builder might go the way of using metal channels and associated bracketry made by ServoCity/Actobotics or similar suppliers as structure for their larger robot. Many manufacturers have used these popular structural components for several iterations of prototypes, but eventually desire to go with bending sheet metal or using machined metal parts for a final product. Only a few experimental robot builders ever get to this point. Most experimenters cut sheet metal and attach pieces together with brackets or by welding to form final structural components.

This is also the area where a lot of potential manufacturers face design and manufacturing problems and begin to cut back on desired features. Funding becomes an issue and multiple design reviews sometimes turn 'great' robots into 'good' robots.

DRC Robots are Great Examples

My September column highlighted the DRC this past June and the 25 amazing entrants. Though six of the entrants used previously-developed million-dollar plus Atlas platforms, there was tremendous ingenuity evident in the various robots. Team Tartan's entry, CHIMP (shown in **Figure 10**) used rollers on its four appendages as testing proved that basic walking might be difficult in a disaster area. Yes, it was very problematic for all the entrants. The winner — KAIST's modified Hubo — did the same thing with wheels on its lower extremities. Team GRIT, Team Robosimian, and Team Nimbro Rescue also deviated from a basic bipedal humanoid. Why? Because walking is really difficult for robots in uncertain environments.

Marc Raibert of Boston Dynamics (who I mentioned earlier) has long been the world's expert on legged robots. I first saw his one-legged "Raibert Hopper" back in '81 or '82 that's shown in **Figure 11**. It was sitting on a rack in his lab, but I saw a video of it hopping and it literally blew my mind. One leg! The pneumatic 'robot' had to keep hopping just to keep from falling over, but it was an excellent example of balancing techniques — long before we had the MEMS accelerometers and gyros of today. Raibert and his team went on to design Big Dog, a robot cheetah, much of the Atlas robot's systems, and other legged robots.

Final Thoughts

The examples I've used here are to illustrate the fact that we can do it — just like that WWII poster of the woman in the aircraft factory. We can build great robots. It may be hard, but we will ultimately be able to create any type of robot as long as there is a true desire.

Even though the mechanical aspect of robot design and production appears to be the biggest stumbling block, folks like Raibert and competitions like the DRC have shown that the robot experts of today will design and build the robots of tomorrow. Better batteries, sensors, and software will happen. There are many untapped applications just waiting for a robotic solution.

So, ignore the naysayers, stick your neck out, and build a robot that can actually do something. Not just wobble around. **SV**

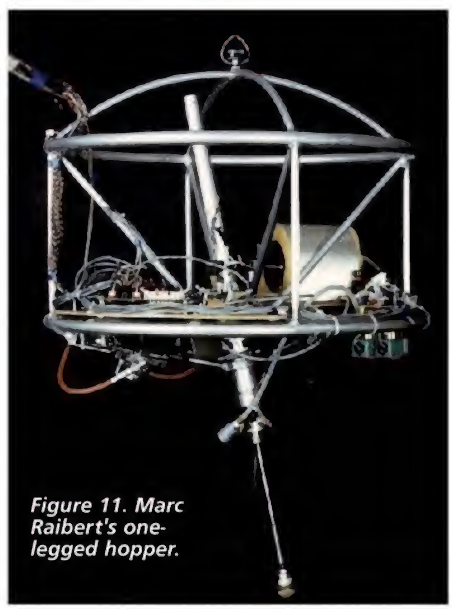


Figure 11. Marc Raibert's one-legged hopper.

ROBO-LINKS

For the finest in robots, parts, and services, go to www.servomagazine.com and click on Robo-Links.

Did You Know Preferred Subscribers get access to all the digital back issues of *SERVO Magazine* for free?

Call for details

1-877-525-2539

ADVERTISER INDEX

All Electronics Corp. 20
AndyMark 80
CES 2016 46
Firgelli 9
PCBCart 3
Hitec 2, 80
littleBits 13
Mikronauts 55
Maxbotix 80
PanaVise 55
Parallax 41
Pololu 82
Ramsey 80
RobotShop, Inc 80, 81
ServoCity 80, 83
Sphero ....Back Cover
SuperBrightLEDs 80
The Robot Marketplace 15
Tormach 31

RobotShop

ROBOTSHOP ACCELERATOR

FOR INVENTORS & DEVELOPERS

ENTREPRENEURS | INNOVATORS / DEVELOPERS | STUDENTS | UNIVERSITIES | STARTUPS AND SMALL COMPANIES

SUBMIT YOUR PROJECT TODAY

With our excellent visibility and worldwide appeal, RobotShop is without a doubt the best partner for commercializing and distributing your robotic inventions.

1 CONTACT US

Begin the process by contacting our engineering team via the RobotShop Support Center.

2 OPTIMIZATION

Our engineers are specialized in simplification and optimization of technologies with the goal of mass commercialization.

3 MANUFACTURING

RobotShop pairs your product with the most suitable manufacturer(s) and takes care of all the overhead.

4 DISTRIBUTION

RobotShop invests in the initial stock with the intent to sell the product on our websites via our global distribution logistics.

5 MAKE MONEY

Finally you will receive a percentage of every sale.

FOR MORE INFORMATION:

robotshop.com/accelerator